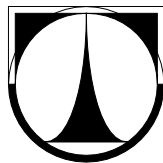


**TECHNICKÁ UNIVERZITA V LIBERCI**  
Fakulta mechatroniky, informatiky a mezioborových studií



**BAKALÁŘSKÁ PRÁCE**

Liberec 2011

**David Botka**

---

# **TECHNICKÁ UNIVERZITA V LIBERCI**

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2612 – Elektrotechnika a informatika

Studijní obor: B2646 – Informační technologie

## **Konverze monofonních audio signálů na stereofonní pomocí NMF**

### **Mono-to-Stereo Conversion of Audio Signals Using Non-Negative Matrix Factorization**

#### **Bakalářská práce**

Autor:	<b>David Botka</b>
Vedoucí práce:	Ing. Zbyněk Koldovský, Ph.D.
Konzultant:	Ing. Jiří Málek

**V Liberci 14. 5. 2011**

STRANA S ORIGINALNÍM ZADÁNÍM

# Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé bakalářské práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé bakalářské práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své bakalářské práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum: 14. 5. 2011

Podpis

## Poděkování

Chtěl bych poděkovat všem, kteří mi pomohli s vypracováním bakalářské práce. Děkuji především vedoucímu mé práce Ing. Zbyňku Koldovskému, Ph.D. za cenné rady, konzultace a pomoc při tvorbě této bakalářské práce.

## Abstrakt

Tato práce pojednává o převodu monofonních audio signálů na stereofonní. Pro převod monofonního audio signálu na stereofonní je třeba rozložit monofonní signál do jednotlivých složek. K tomu využíváme metodu non-negative matrix factorization (NMF). Popisujeme její vlastnosti, možnosti využití a iterační algoritmy výpočtu. K měření kvality rozkladu používáme hodnoty signal to interference ratio (SIR) a signal to distortion ratio (SDR). Rozklady provádíme na signálech pomocí NMF se třemi různými objektivními funkcemi. Porovnáváme výsledky dosažené s Euklidovskou vzdáleností, Kullback-Leibler (KL) divergencí a Itakura-Saito (IS) divergencí. Dále popisujeme digitální signál v časové a frekvenční oblasti a navrhujeme postup převodu monofonního signálu na stereofonní.

Cílem práce bylo navrhnout a popsat postup pro převod monofonního audio signálu na stereofonní a pro realizaci implementovat algoritmy v prostředí MATLAB. Dále pak porovnat výsledky NMF rozkladu s různými objektivními funkcemi a z výsledků rozkladu vytvořit stereofonní signál.

**Klíčová slova:** non-negative matrix factorization (NMF), převod monofonních audio signálů na stereofonní, signal to interference ratio (SIR), signal to distortion ratio (SDR), spektrogram

## Abstract

This thesis discusses the mono to stereo conversion of audio signals. For mono to stereo conversion is necessary to separate components from mono signal. For separation we apply a method called non-negative matrix factorization (NMF). We describe its characteristics, possibility of using and update rules for computation. To measure the quality of separation we use the signal to interference ratio (SIR) and the signal to distortion ratio (SDR). We perform separations of components from audio signals using NMF with three different cost functions. We compare results reached with Euclidean distance, Kullback-Leibler (KL) divergence and Itakura-Saito (IS) divergence. Next we describe digital signal in time and frequency domain and propose the process for mono to stereo conversion of audio signals.

The aim of the thesis was to propose and describe the process for mono to stereo conversion of audio signals and for realization to implement algorithms in the MATLAB environment. Then to compare results reached with different cost functions and to make stereo signal from results of separation.

**Key words:** non-negative matrix factorization (NMF), mono to stereo conversion of audio signals, signal to interference ratio (SIR), signal to distortion ratio (SDR), spectrogram

# Obsah

Prohlášení . . . . .	iii
Poděkování . . . . .	iv
Abstrakt . . . . .	v
Abstract . . . . .	vi
Obsah . . . . .	vii
Seznam obrázků . . . . .	ix
Seznam tabulek . . . . .	x
Seznam zkratek . . . . .	xi
<b>1 Úvod</b>	<b>1</b>
<b>2 Non-negative matrix factorization</b>	<b>3</b>
2.1 Historie . . . . .	4
2.2 Podobné matrix factorization metody . . . . .	4
2.3 Podmínka nezápornosti . . . . .	4
2.4 Oblasti použití . . . . .	5
2.5 Výpočet . . . . .	6
2.6 Objektivní funkce . . . . .	6
2.7 Multiplikativní iterační algoritmus . . . . .	7
2.8 Aditivní iterační algoritmus . . . . .	9
<b>3 Popis digitálního signálu</b>	<b>10</b>
3.1 Časová oblast . . . . .	10
3.2 Frekvenční oblast . . . . .	11
3.2.1 Diskrétní Fourierova transformace . . . . .	11
3.2.2 Okénkovací funkce . . . . .	12



3.2.3	Amplitudové spektrum . . . . .	13
3.2.4	Spektrogram . . . . .	14
<b>4</b>	<b>Převod monofonního signálu na stereofonní</b>	<b>17</b>
4.1	Vytvoření masky . . . . .	18
4.1.1	Binární maska . . . . .	18
4.2	Inverzní spektrogram . . . . .	19
4.3	Vytvoření stereo signálu . . . . .	21
<b>5</b>	<b>Měření kvality rozkladu – SIR a SDR</b>	<b>22</b>
5.1	Postup výpočtu . . . . .	23
<b>6</b>	<b>Praktická aplikace</b>	<b>25</b>
6.1	Rozklad signálu složeného ze tří složek . . . . .	26
6.1.1	Zhodnocení výsledků . . . . .	29
6.2	Rozklad zašuměného signálu . . . . .	31
6.2.1	Zhodnocení výsledků . . . . .	32
6.3	Rozklad reálné nahrávky . . . . .	33
6.3.1	Zhodnocení výsledků . . . . .	34
<b>7</b>	<b>Závěr</b>	<b>36</b>
<b>A</b>	<b>Obsah přiloženého CD</b>	<b>39</b>
<b>B</b>	<b>Manuál k implementaci</b>	<b>40</b>
B.1	NMF . . . . .	40
B.2	Maska . . . . .	41
B.3	Inverzní spektrogram . . . . .	42
B.4	Uložení výsledků . . . . .	43
B.5	Smíchání zvukových signálů . . . . .	44
B.6	Výpočet SIR . . . . .	45
B.7	Výpočet SDR . . . . .	46
	<b>Literatura</b>	<b>47</b>

# Seznam obrázků

3.1	Digitální signál v základní formě a v normalizované formě . . . . .	10
3.2	Okénkovací funkce . . . . .	12
3.3	Aplikace okénkovací funkce na signál . . . . .	13
3.4	Dvoustranné a jednostranné amplitudové spektrum . . . . .	13
3.5	Skutečné a rozmazané amplitudové spektrum . . . . .	14
3.6	Grafické znázornění výpočtu spektrogramu . . . . .	14
3.7	Spektrogram záznamu hry na bicí . . . . .	15
3.8	Vliv velikosti bloku na spektrogram . . . . .	16
3.9	Vliv velikosti překrytí bloků na spektrogram . . . . .	16
4.1	Grafické znázornění převodu monofonního signálu na stereofonní . . .	17
4.2	Grafické znázornění výsledků NMF . . . . .	18
4.3	Aplikace binární masky na spektrogram . . . . .	19
4.4	Grafické znázornění převodu spektrogramu do časové oblasti . . . . .	19
4.5	Určení pomocného vektoru okénkovacích funkcí . . . . .	20
6.1	Složky signálu $S_1$ , $S_2$ , $S_3$ a součet $X$ v časové oblasti a jejich spektro- gramy . . . . .	26
6.2	Původní monofonní a výsledný stereofonní signál v časové oblasti . .	31
6.3	Původní monofonní a výsledný stereofonní signál v časové oblasti . .	33
6.4	Původní monofonní a výsledný stereofonní signál v časové oblasti . .	35

# Seznam tabulek

6.1	Výsledky rozkladu do tří složek . . . . .	27
6.2	Výsledky rozkladu do čtyř složek . . . . .	27
6.3	Výsledky rozkladu do pěti složek . . . . .	28
6.4	Výsledky rozkladu do šesti složek . . . . .	28
6.5	Hodnoty SIR a SDR rozkladu signálu složeného ze tří složek . . . . .	30
6.6	Časy výpočtů rozkladu signálu složeného ze tří složek . . . . .	30
6.7	Hodnoty SIR a SDR rozkladu zašuměného signálu . . . . .	32
6.8	Časy výpočtů rozkladu části písně <i>Hledaný muž</i> od kapely <i>Taxmeni</i> .	34

# Seznam zkratek

<b>ADC</b>	Analogově-digitální převodník
<b>DFT</b>	Diskrétní Fourierova transformace
<b>EUC</b>	Euklidovská vzdálenost
<b>FFT</b>	Fast Fourier transform
$f_s$	vzorkovací frekvence
<b>ICA</b>	Independent component analysis
<b>IDFT</b>	Inverzní diskrétní Fourierova transformace
<b>IS</b>	Itakura-Saito
<b>KL</b>	Kullback-Leibler
<b>LDA</b>	Linear discriminant analysis
<b>NBITS</b>	počet bitů kvantizačních úrovní
<b>NMF</b>	Non-negative matrix factorization
<b>PMF</b>	Positive matrix factorization
<b>SCA</b>	Sparse component analysis
<b>SDR</b>	Signal to distortion ratio
<b>SIR</b>	Signal to interference ratio
<b>STFT</b>	Short-time Fourierova transformace
<b>PCA</b>	Principal component analysis

# Kapitola 1

## Úvod

Audio signály byly dříve zaznamenávány a tudíž i reprodukovány pouze jako monofonní. S rozvojem technologie se začaly používat signály stereofonní, které mají oproti monofonním signálům určité výhody. Při přehrávání monofonního audio signálu na soustavě s více reproduktory půjde ze všech reproduktorů stejný zvuk. V případě stereofonního audio signálu může každý reproduktor přehrávat jiný kanál. Pokud budou reproduktory správně rozmístěny, vznikne prostorový efekt, který obvykle vyvolává v posluchačích výrazně lepší dojem než při poslechu monofonního audio signálu.

Některé zvukové signály jsou pořízeny nebo uloženy jen jako monofonní. Existuje ale způsob, kterým se může povést tyto monofonní audio signály převést na stereofonní. Audio signál představující hudbu bývá složen z více složek, kde jednu složku může představovat hudební nástroj, nízké frekvence, vysoké frekvence, zpěv atd. Smysl stereofonních signálů spočívá v tom, že ve všech kanálech není stejný signál. Každý kanál stereofonního signálu může obsahovat různé složky, zatímco monofonní signál má vždy vše smíchané v jenom kanálu a jednotlivé složky nejsou obvykle k dispozici zvlášť.

Pro převod monofonního audio signálu na stereofonní je hlavním a nejobtížnějším krokem oddělení jednotlivých složek ze zpracovávaného signálu. K tomu lze využít metodu nazývanou non-negative matrix factorization (NMF), kterou představili Lee a Seung [1] v roce 1999.

Metoda non-negative matrix factorization (NMF) předpokládá, že celek se skládá z jednotlivých složek. Cílem je tyto složky detekovat – najít, co patří k sobě. Pokud tento proces dopadne úspěšně, je pak možné jednotlivé složky od sebe oddělit a z nich vytvořit stereofonní signál. Oddělené samostatné složky pak už stačí rozmístit do různých kanálů. Při rozkladu hudby lze předpokládat, že výsledné složky budou představovat jednotlivé hudební nástroje. Kvalita výsledného stereofonního signálu závisí především na kvalitě rozkladu do jednotlivých složek. Pokud se žádná složka nepodaří ani částečně oddělit, výsledný stereofonní signál nemusí znít příliš dobře. Rozmístit vyseparované složky do kanálů lze pak například podle hlasitosti, obsažených frekvencí nebo i podle hudebních nástrojů.

Pro NMF bylo nalezeno uplatnění při řešení různých problémů v mnoha oborech. Jedná se například o analyzování, klastrování i rozpoznávání. Pomocí NMF lze zpracovávat různé typy dat, například obrázky, texty i zvuky.

## Kapitola 2

# Non-negative matrix factorization

Non-negative matrix factorization je v současnosti populární metoda s širokým využitím. Pro vstupní matici  $\mathbf{V}$  s rozměry  $n \times m$  obsahující pouze nezáporná data je cílem najít rozklad

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} \quad (2.1)$$

s podmínkou, že matice  $\mathbf{W}$  i  $\mathbf{H}$  musí být také nezáporné. Matice  $\mathbf{W}$  má rozměry  $n \times r$  a matice  $\mathbf{H}$  má rozměry  $r \times m$ , kde  $r$  je redukovaná dimenze matic. Redukovaná dimenze matic  $r$  je volitelný parametr a udává, do kolika složek se provede rozklad. Obecně se volí tak, aby byla splněna nerovnost (2.2).

$$r < \frac{n \cdot m}{n + m} \quad (2.2)$$

Při splnění nerovnosti (2.2) dochází k redukci dat a výsledek součinu matic  $\mathbf{W}$  a  $\mathbf{H}$  může být chápán jako komprimovaná forma matice  $\mathbf{V}$  (Lee a Seung [1]). Jak vyjadřuje vztah (2.1), součin matic  $\mathbf{W}$  a  $\mathbf{H}$  obecně představuje jen aproximaci matice  $\mathbf{V}$ , proto se NMF někdy označuje jako *approximate non-negative matrix factorization* nebo *non-negative matrix approximation* (Févotte a kol. [2]).

Rozklad pomocí NMF by měl být jednoznačný, ale pořadí výsledných složek se předem pokládá za nejednoznačné. Kvůli této vlastnosti může být NMF pro některé praktické aplikace nepoužitelná.

## 2.1 Historie

Paatero a Tapper [3] původně navrhli metodu s názvem *positive matrix factorization* (PMF). Později Lee a Seung [4] ve svém prvním článku s touto problematikou nazývali tuto metodu *conic coding*. V jejich dalších pracích se již objevuje název *non-negative matrix factorization*. Následně v článku [5] představili jednoduchý multiplikativní iterační algoritmus pro výpočet NMF. Od té doby vzniklo mnoho modifikací, rozšíření a zobecnění a NMF se tak dá využít pro různé účely (Schmidt [6]).

## 2.2 Podobné matrix factorization metody

Vedle NMF existují i další metody řešící podobnou problematiku. Mezi ně patří například *principal component analysis* (PCA), *independent component analysis* (ICA), *sparse component analysis* (SCA) a *linear discriminant analysis* (LDA). Všechny tyto metody se vzájemně liší, a proto i pro stejná vstupní data poskytují různé výsledky. Nelze ale samozřejmě některou metodu označit za nejlepší, protože pro řešení různých problémů může být vhodnější využít metodu jinou.

## 2.3 Podmínka nezápornosti

Mnohá data popisující určitý celek jsou v principu nezáporná. Například amplitudové spektrum, počet výskytů i obrazová data, kde barvy jednotlivých pixelů mohou být číselně vyjádřeny pomocí zastoupení jednotlivých barevných složek.

Skládání celku z jednotlivých nezáporných částí navíc umožňuje pouze aditivní kombinace, žádné vzájemné vyrušení nemůže nastat. Tato vlastnost se i shoduje s intuitivním přístupem, že celek je součtem svých částí. Nezápornost všech matic je tedy u NMF důvodná podmínka (Schmidt [6]).



## 2.4 Oblasti použití

Možnosti využití NMF, jejího zobecnění a rozšíření jsou velmi široké. Jedná se například o redukci dimenze, hledání příznaků, klastrování, oddělení složek z mixu a i rozpoznávání. Možné oblasti použití popsal Schmidt [6]:

- Zpracování obrazu
  - hledání částí obličeje v obrázku
  - rozpoznávání obličeje
  - kódování řídkého obrazu
  - hledání krátkých video sekvencí, které reprezentují celý video záznam
- Zpracování textu
  - hledání sémanticky podobných slov
  - rozpoznávání jazyka
- Bioinformatika
  - analýza dat popisujících geny za účelem rozlišení různých druhů rakoviny
  - klasifikace EEG signálu
  - analýza chemických změn v lidském mozku
  - zpracování dat z pozitronové emisní tomografie vyjadřující srdeční činnost
- Zpracování zvuku
  - přepis polyfonní hudby
  - hledání spektrálních vlastností pro klasifikaci zvukových signálů
  - oddělení zdrojů zvuku z mixu
- Ostatní
  - analýza astronomických dat
  - analýza barevného spektra

## 2.5 Výpočet

Rozklad matic (2.1) je obvykle hledán jako úloha minimalizace

$$\min_{\mathbf{W}, \mathbf{H} \geq 0} D(\mathbf{V} | \mathbf{WH}), \quad (2.3)$$

kde  $D(\mathbf{V} | \mathbf{WH})$  je objektivní funkce, která určuje kvalitu aproximace. Taková objektivní funkce může být vytvořena jako měření rozdílnosti dvou matic. Cílem je tedy najít takové matice  $\mathbf{W}$  a  $\mathbf{H}$ , aby rozdílnost vstupní matice  $\mathbf{V}$  a aproximace  $\mathbf{WH}$  byla co nejmenší (Lee a Seung [5]).

## 2.6 Objektivní funkce

Objektivní funkce je definovaná jako

$$D(\mathbf{V} | \mathbf{WH}) = \sum_{i=1}^n \sum_{j=1}^m d([\mathbf{V}]_{i,j} | [\mathbf{WH}]_{i,j}), \quad (2.4)$$

kde  $d(x|y)$  je skalární objektivní funkce. Způsobů, jak počítat hodnotu skalární objektivní funkce, je více. Často se používá Euklidovská vzdálenost definovaná vztahem (2.5)

$$d_{EUC}(x | y) = \frac{1}{2}(x - y)^2 \quad (2.5)$$

a zobecněná Kullback-Leibler (KL) divergence<sup>1</sup>, někdy označovaná jako I-divergence definovaná vztahem (2.6)

$$d_{KL}(x | y) = x \cdot \log \frac{x}{y} - x + y. \quad (2.6)$$

---

<sup>1</sup>Euklidovská vzdálenost (2.5) je symetrická, tzn.  $d(x|y) = d(y|x)$ . V případě KL (2.6) se ale obecně  $d(x|y) \neq d(y|x)$ , z toho důvodu to nelze nazývat vzdáleností. Používá se proto pojem divergence.

Další možnou divergencí je Itakura-Saito (IS) divergence definovaná vztahem (2.7).

$$d_{IS}(x \mid y) = \frac{x}{y} - \log \frac{x}{y} - 1 \quad (2.7)$$

$\beta$ -divergence je definovaná vztahem (2.8).

$$d_{\beta}(x \mid y) = \begin{cases} \frac{1}{\beta(\beta-1)}(x^{\beta} + (\beta-1)y^{\beta} - \beta xy^{\beta-1}) & \beta \in \mathbb{R} \setminus \{0, 1\} \\ x(\log(x) - \log(y)) + (y - x) & \beta = 1 \\ \frac{x}{y} - \log \frac{x}{y} - 1 & \beta = 0 \end{cases} \quad (2.8)$$

IS divergence je limitní případ  $\beta$ -divergence pro  $\beta = 0$ . Podobně i KL divergence pro  $\beta = 1$  a Euklidovská vzdálenost pro  $\beta = 2$ .

Existuje i řada dalších divergencí použitelných pro vytvoření objektivní funkce, například Cichocki a kol. [7] navrhli algoritmy s využitím Csiszár divergencí a Amariho  $\alpha$ -divergence. Dhillon a Sra [8] popsali algoritmy pro širokou rodinu Bregmanových divergencí.

NMF s různými objektivními funkcemi poskytuje různé výsledky a výběr objektivní funkce by měl být proveden na základě typu analyzovaných dat.

## 2.7 Multiplikativní iterační algoritmus

Využití Euklidovské vzdálenosti (2.5) a Kullback-Leibler (KL) divergence (2.6) pro výpočet NMF původně navrhli Lee a Seung [5]. Jejich odvození iteračního algoritmu je založeno na minimalizaci (2.3) pomocí metody *gradient descent*.

Multiplikativní iterační algoritmus s Euklidovskou vzdáleností je popsán vztahy (2.9) a (2.10).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} \frac{(\mathbf{W}^T \mathbf{V})_{a\mu}}{(\mathbf{W}^T \mathbf{W} \mathbf{H})_{a\mu}} \quad (2.9)$$

$$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{(\mathbf{W} \mathbf{H}^T)_{ia}}{(\mathbf{W} \mathbf{H} \mathbf{H}^T)_{ia}} \quad (2.10)$$

Multiplikativní iterační algoritmus s KL divergencí je popsán vztahy (2.11) a (2.12).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} \frac{\sum_i \mathbf{W}_{ia} \mathbf{V}_{i\mu} / (\mathbf{WH})_{i\mu}}{\sum_k \mathbf{W}_{ka}} \quad (2.11)$$

$$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{\sum_\mu \mathbf{H}_{a\mu} \mathbf{V}_{i\mu} / (\mathbf{WH})_{i\mu}}{\sum_v \mathbf{H}_{av}} \quad (2.12)$$

Lee a Seung [5] dále dokázali, že objektivní funkce  $D(\mathbf{V}|\mathbf{WH})$  je se zmíněnými iteračními algoritmy nerostoucí. To znamená, že než  $D(\mathbf{V}|\mathbf{WH})$  zkonverguje k minimu, s každým krokem se výsledek o něco zlepší. Iterační algoritmy ale nezaručují konvergenci do globálního minima, pouze do lokálního.

IS a  $\beta$ -divergenci popsali Févotte a kol. [2]. Multiplikativní iterační algoritmus pro NMF s IS divergencí vyjadřují vztahy (2.13) a (2.14).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} \frac{(\mathbf{W}^T((\mathbf{WH})^{[-2]} \cdot \mathbf{V}))_{a\mu}}{(\mathbf{W}^T(\mathbf{WH})^{[-1]})_{a\mu}} \quad (2.13)$$

$$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{(((\mathbf{WH})^{[-2]} \cdot \mathbf{V})\mathbf{H}^T)_{ia}}{((\mathbf{WH})^{[-1]}\mathbf{H}^T)_{ia}} \quad (2.14)$$

Multiplikativní iterační algoritmus pro  $\beta$ -divergenci je popsán vztahy (2.15) a (2.16).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} \frac{(\mathbf{W}^T((\mathbf{WH})^{[\beta-2]} \cdot \mathbf{V}))_{a\mu}}{(\mathbf{W}^T(\mathbf{WH})^{[\beta-1]})_{a\mu}} \quad (2.15)$$

$$\mathbf{W}_{ia} \leftarrow \mathbf{W}_{ia} \frac{(((\mathbf{WH})^{[\beta-2]} \cdot \mathbf{V})\mathbf{H}^T)_{ia}}{((\mathbf{WH})^{[\beta-1]}\mathbf{H}^T)_{ia}} \quad (2.16)$$

Jak již bylo zmíněno, pro  $\beta = 2$  (Euklidovská vzdálenost) a pro  $\beta = 1$  (KL divergence) je dokázáno, že objektivní funkce  $D(\mathbf{V}|\mathbf{WH})$  je nerostoucí (Lee a Seung [5]). Kompass [9] tento důkaz zobecnil a ukázal, že  $D(\mathbf{V}|\mathbf{WH})$  je nerostoucí pro  $1 \leq \beta \leq 2$ . V praxi se ukazuje, že  $D(\mathbf{V}|\mathbf{WH})$  je nerostoucí i pro  $\beta < 1$  a pro  $\beta > 2$ , tudíž i pro  $\beta = 0$  (IS divergence), důkaz ale nebyl zatím nalezen (Févotte a kol. [2]).

## 2.8 Aditivní iterační algoritmus

Vedle multiplikativních iteračních algoritmů pro výpočet NMF existují i aditivní iterační algoritmy. Lee a Seung [5] navrhli také aditivní iterační algoritmus s Euklidovskou vzdáleností (2.5) a KL divergencí (2.6).

Aditivní iterační algoritmus s Euklidovskou vzdáleností popisuje vztah (2.17).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} + \eta_{a\mu} [(\mathbf{W}^T \mathbf{V})_{a\mu} - (\mathbf{W}^T \mathbf{W} \mathbf{H})_{a\mu}] \quad (2.17)$$

Aditivní iterační algoritmus s KL divergencí popisuje vztah (2.18).

$$\mathbf{H}_{a\mu} \leftarrow \mathbf{H}_{a\mu} + \eta_{a\mu} \left[ \sum_i \mathbf{W}_{ia} \frac{\mathbf{V}_{i\mu}}{(\mathbf{W} \mathbf{H})_{i\mu}} - \sum_i \mathbf{W}_{ia} \right] \quad (2.18)$$

Pokud je v případě Euklidovské vzdálenosti  $\eta_{a\mu}$  nastaveno na

$$\eta_{a\mu} = \frac{\mathbf{H}_{a\mu}}{(\mathbf{W}^T \mathbf{W} \mathbf{H})_{a\mu}} \quad (2.19)$$

dostaneme stejný algoritmus, jaký udává vztah (2.9). Podobně i pro KL divergenci, pokud je  $\eta_{a\mu}$  nastaveno na

$$\eta_{a\mu} = \frac{\mathbf{H}_{a\mu}}{\sum_i \mathbf{W}_{ia}} \quad (2.20)$$

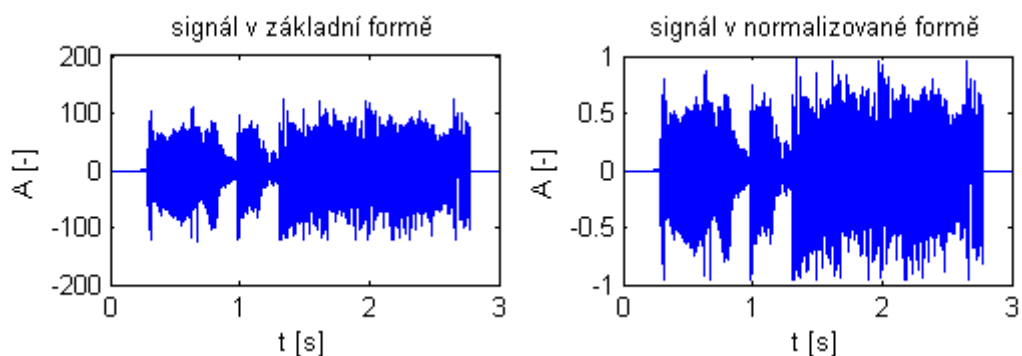
dostaneme stejný algoritmus jako ve vztahu (2.11).

# Kapitola 3

## Popis digitálního signálu

### 3.1 Časová oblast

Digitální signál je v časové oblasti řada čísel, která může vzniknout z naměřených nebo vypočtených hodnot, vygenerováním s určitými parametry nebo i převodem analogového signálu na digitální pomocí analogově-digitálního převodníku (ADC). Při analogově-digitálním převodu dochází ke vzorkování (snímání hodnot analogového signálu) a kvantování (přiřazení úrovně). Nejdůležitějším parametrem vzorkování je vzorkovací frekvence  $f_s$  a pro kvantování počet možných úrovní  $2^{NBITS}$ . S digitálním signálem se často pracuje v normalizované formě. Pak je rozsah úrovní v intervalu  $\langle -1;1 \rangle$ . Zobrazení digitálního signálu v základní a normalizované formě je na obrázku (3.1).



Obrázek 3.1: Digitální signál v základní formě a v normalizované formě

## 3.2 Frekvenční oblast

Digitální signál lze popsat i ve frekvenční oblasti. Často se používá amplitudové spektrum spolu s fázovým spektrem a spektrogram. Amplitudové a fázové spektrum vyjadřuje závislosti amplitud a fází na frekvenci a spektrogram znázorňuje zastoupení jednotlivých frekvencí v závislosti na čase.

K výpočtu spektra digitálních signálů se využívá diskrétní Fourierova transformace (DFT).

### 3.2.1 Diskrétní Fourierova transformace

Fourierova transformace umožňuje rozklad periodického signálu na jednotlivé harmonické složky. Při práci se vzorkovaným signálem se používá diskrétní Fourierova transformace (DFT) popsaná vztahem (3.1).

$$X[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi nk/N} \quad (3.1)$$

$N$  ..... počet vybraných vzorků (předpoklad - jde o jednu periodu)

$X[k]$  ...  $k$ -tý koeficient z  $N$  vzorků

$x[n]$  ...  $n$ -tý vzorek signálu

Výstupem je  $N$  komplexních koeficientů diskrétního spektra s hodnotami na frekvencích  $k \cdot f_s/N$ . Pro reálné signály stačí ale vypočítat jen  $N/2$  hodnot, protože ostatní hodnoty jsou k nim komplexně sdružené. Výsledky pro  $k > N$  jsou stejné jako pro základní interval  $-N/2 < k < N/2$ , protože spektrum je periodické. Pokud vybraných  $N$  vzorků nepředstavuje jednu periodu, výsledné spektrum je zatížené různými chybami a jedná se pouze o aproximaci spektra. Může dojít například k rozmazání spektra (objeví se neexistující složky).

V praxi se obvykle používá optimalizovaný výpočet DFT nazývaný FFT (Fast Fourier Transform), který poskytuje stejné výsledky s výrazně nižší výpočetní náročností.

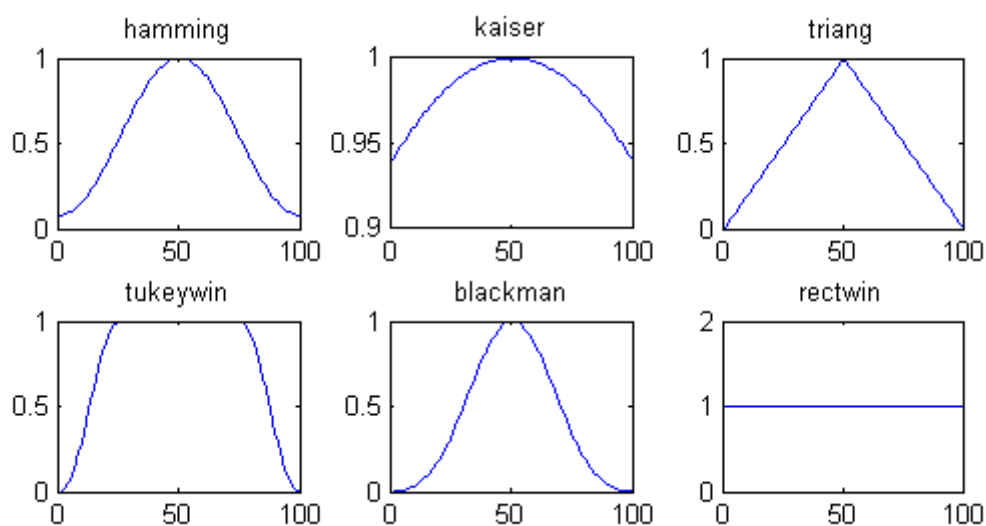
K diskrétní Fourierově transformaci existuje také inverzní operace. Inverzní diskrétní Fourierova transformace (IDFT) je popsána vztahem (3.2).

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cdot e^{j2\pi nk/N} \quad (3.2)$$

Rozdíl mezi vztahem pro DFT a IDFT je jen ve znaménku v exponenciální funkci. Vstupem IDFT musí být vždy  $N$  komplexních koeficientů dvoustranného spektra, nestačí jen  $N/2$  hodnot jednostranného spektra. U reálných signálů se častěji pracuje s jednostranným spektrem. Pro aplikaci IDFT je třeba z jednostranného spektra nejprve určit spektrum dvoustranné (Nouza [10]).

### 3.2.2 Okénkovací funkce

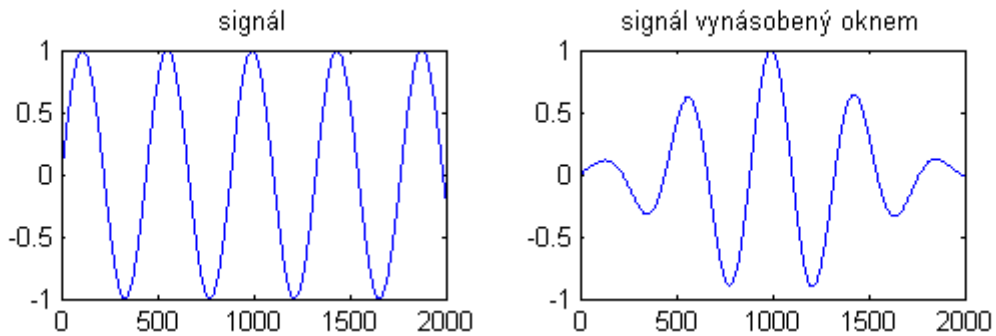
Zpracovávané signály nemusí být periodické nebo jejich perioda není známa, a proto dochází k rozmazání spektra téměř vždy. Vynásobením výřezu signálu okénkovací funkcí lze alespoň částečně zamezit rozmazání spektra. Některé okénkovací funkce jsou na obrázku (3.2).



Obrázek 3.2: Okénkovací funkce



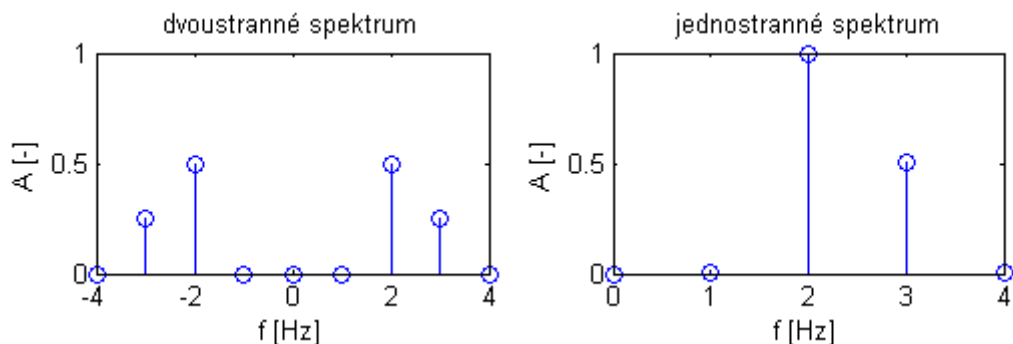
Signál vynásobený Hammingovou okénkovací funkcí zobrazuje obrázek (3.3).



Obrázek 3.3: Aplikace okénkovací funkce na signál

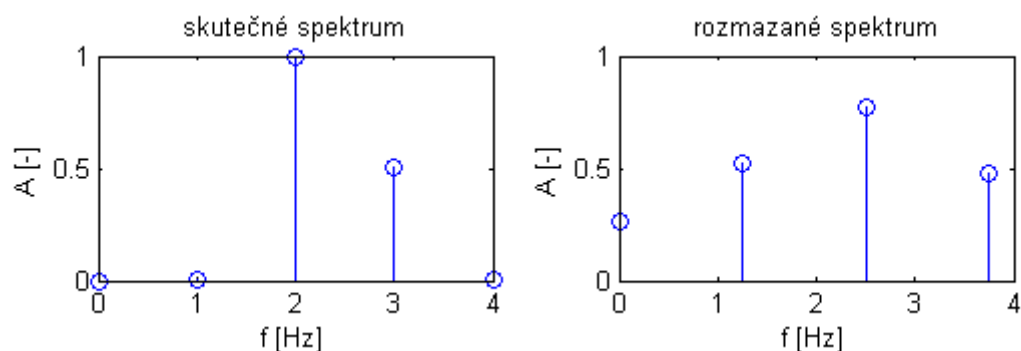
### 3.2.3 Amplitudové spektrum

Amplitudové spektrum zobrazuje v signálu obsažené frekvence a jejich amplitudy. Může být jednostranné nebo dvoustranné. Výsledkem většiny výpočetních postupů je dvoustranné spektrum, ale v praxi se u reálných signálů častěji používá jednostranné. Pro určení jednostranného spektra z dvoustranného se vezme jeho polovina a amplitudy se vynásobí dvěma. Obrázek (3.4) zobrazuje dvoustranné a jednostranné amplitudové spektrum signálu, který je složen z dvou kosínusových průběhů. První má frekvenci 2 Hz s amplitudou 1 a druhý frekvenci 3 Hz s amplitudou 0.5.



Obrázek 3.4: Dvoustranné a jednostranné amplitudové spektrum

Při výpočtu amplitudového spektra může dojít v důsledku různých chyb k jeho rozmazání, tzn. objeví se v něm neexistující složky. Jedná se pak pouze o aproximaci spektra. Obrázek (3.5) porovnává skutečné a rozmazané spektrum (Nouza [10]).

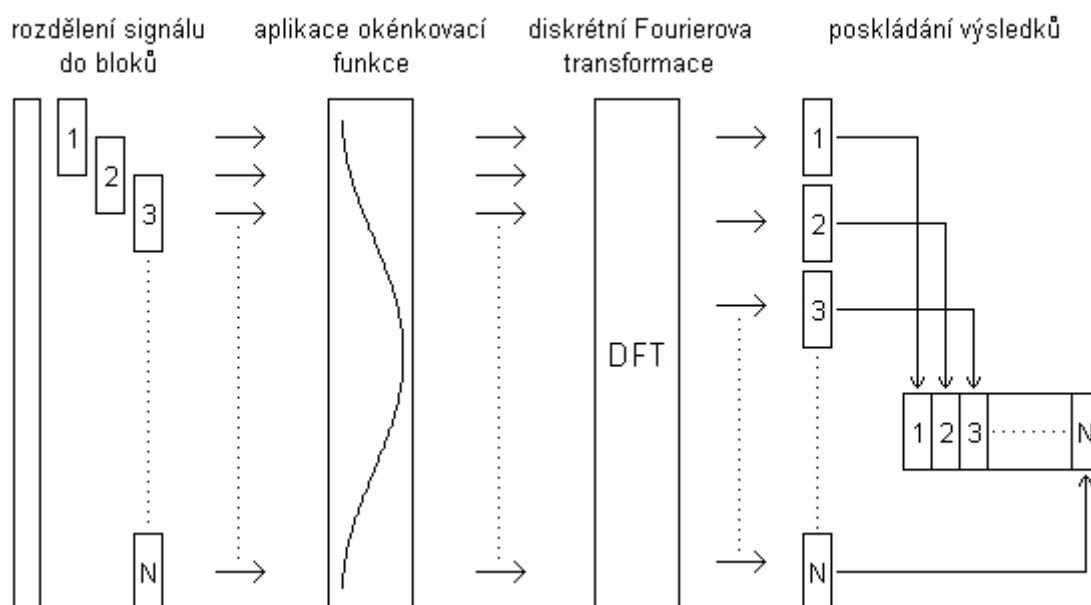


Obrázek 3.5: Skutečné a rozmazané amplitudové spektrum

### 3.2.4 Spektrogram

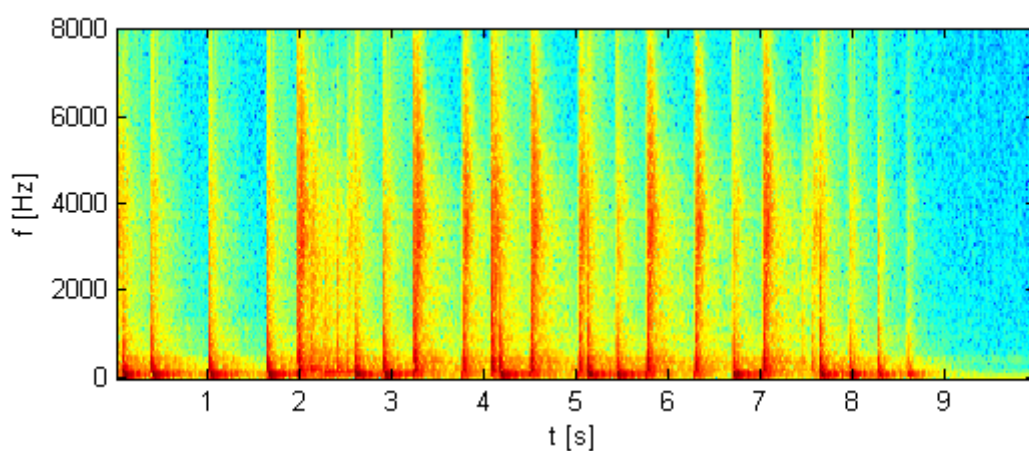
Spektrogram digitálního signálu vyjadřuje zastoupení jednotlivých frekvencí v závislosti na čase. Nejčastěji se zobrazuje jako barevný obrázek, kde na svislé ose je frekvence a na vodorovné ose čas. Čím sytější barva, tím je odpovídající frekvence v daný čas více zastoupená.

K výpočtu spektrogramu se využívá short-time Fourierovy transformace (STFT). Tento proces je znázorněn na obrázku (3.6).



Obrázek 3.6: Grafické znázornění výpočtu spektrogramu

Během výpočtu spektrogramu je signál nejprve rozdělen do stejně dlouhých bloků, přičemž tyto bloky se mohou překrývat. Každý blok se vynásobí okénkovací funkcí (obrázek 3.3) a potom se zvlášť na každý blok aplikuje DFT. Výsledky DFT všech bloků jsou pak poskládány vedle sebe. U reálných signálů se pracuje jen s polovinou výsledků DFT, protože druhá polovina je k první komplexně sdružená a pro spektrogram tak nenese žádnou důležitou informaci. Spektrogram se zobrazuje jako desetinásobek logaritmu absolutních hodnot. Spektrogram záznamu hry na bicí je na obrázku (3.7).

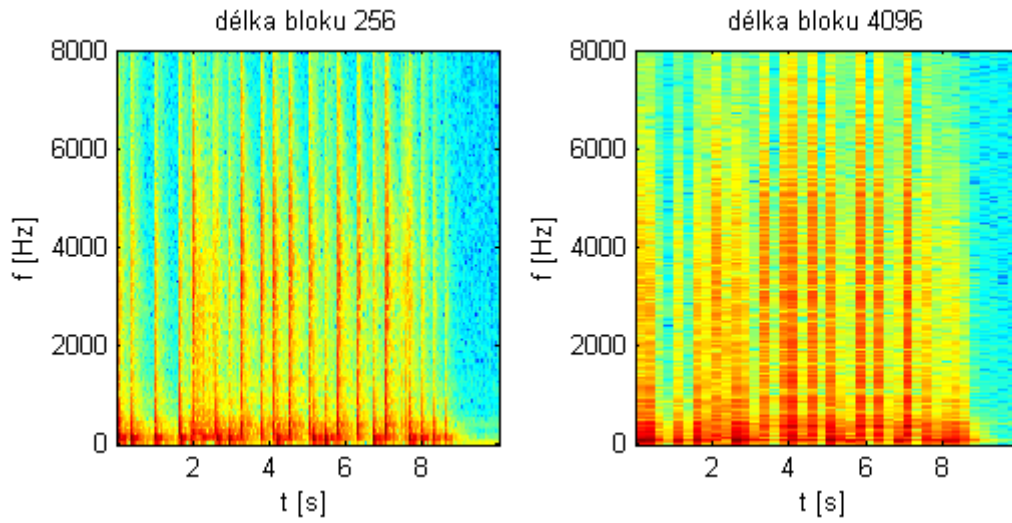


Obrázek 3.7: Spektrogram záznamu hry na bicí

Více spektrogramů pro jeden signál se může od sebe lišit. Záleží na zvolené délce bloků, na okénkovací funkci i na velikosti překrývání bloků. Různé nastavení těchto parametrů přináší určité výhody i nevýhody.

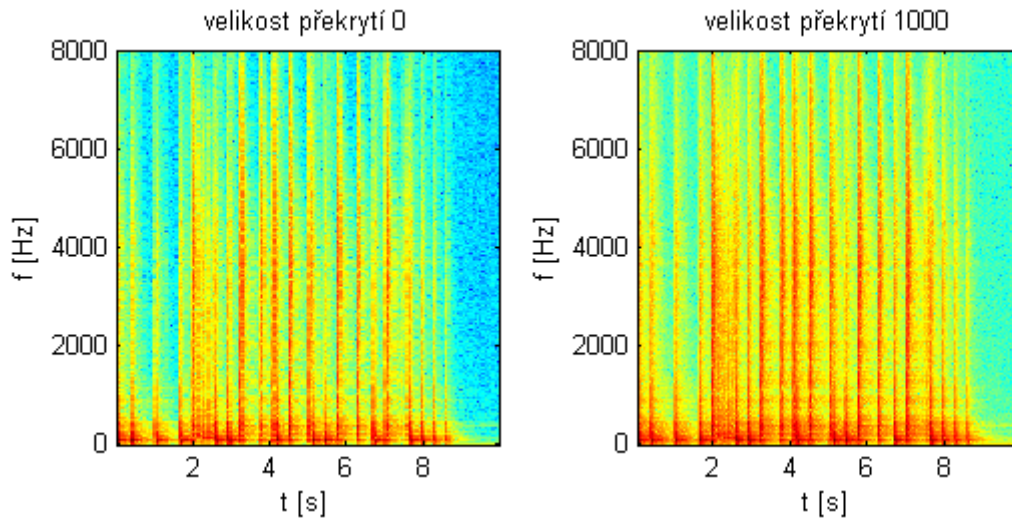
Délka bloků ovlivní přesnost rozlišení v čase a ve frekvenci. S delším blokem bude lepší rozlišení ve frekvenci, ale horší v čase. S kratším blokem naopak – lepší rozlišení v čase, ale horší ve frekvenci. Větší délka bloků také sníží výpočetní náročnost. Dále záleží na velikosti překrývání bloků. S větším překrýváním bloků se dosáhne lepšího popisu signálu, ale výpočetní náročnost se výrazně zvýší. Při výpočtu spektrogramu se kvůli vyhlazení používají okénkovací funkce, tudíž je výsledek ovlivněn i tím. Různými okénkovacími funkcemi se provede vyhlazení jinak [11].

Obrázek (3.8) znázorňuje odlišné výsledky v závislosti na délce bloků. Velikost překrytí bloků je 128 vzorků a použita byla Hammingova okénkovací funkce.



Obrázek 3.8: Vliv velikosti bloku na spektrogram

Obrázek (3.9) porovnává výsledky v závislosti na velikosti překrývání bloků. Délka bloků je 1024 vzorků a použita byla opět Hammingova okénkovací funkce.



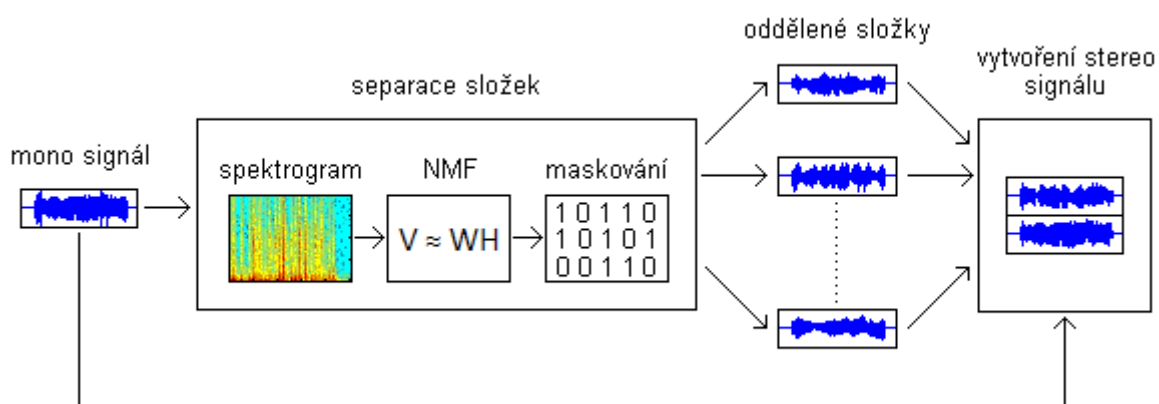
Obrázek 3.9: Vliv velikosti překrytí bloků na spektrogram

## Kapitola 4

# Převod monofonního signálu na stereofonní

Převod monofonního signálu na stereofonní navrhujeme z několika částí. Nejprve vypočítáme amplitudový spektrogram zpracovávaného signálu a na něj pak aplikujeme NMF. Výsledky NMF použijeme k vytvoření masek, kterými pak provedeme samotnou separaci jednotlivých složek. Masky jsou matice se stejnými rozměry jako spektrogram zpracovávaného signálu a separace složky proběhne vynásobením prvek po prvku spektrogramu s maskou. Každá hodnota masky určuje, zda daná frekvence bude v daný čas součástí složky.

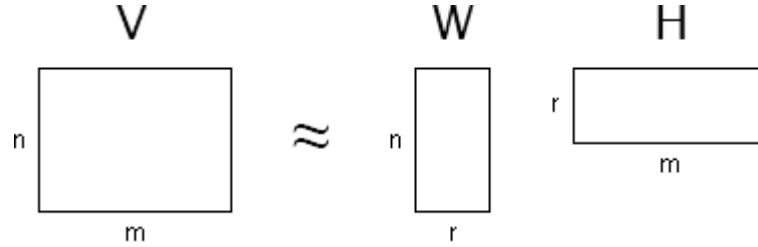
Pokud se podaří některé složky odseparovat, stačí je pak už jen rozdělit do různých kanálů. Grafické znázornění postupu převodu monofonního signálu na stereofonní je na obrázku (4.1).



Obrázek 4.1: Grafické znázornění převodu monofonního signálu na stereofonní

## 4.1 Vytvoření masky

Správné vytvoření masky je velmi důležité pro celý proces separace jednotlivých složek. K vytvoření masky použijeme výsledky NMF, které znázorňuje obrázek (4.2).



Obrázek 4.2: Grafické znázornění výsledků NMF

Aproximace  $\mathbf{WH}$  lze rozepsat na součet maticových násobení sloupců matice  $\mathbf{W}$  a příslušných řádků  $\mathbf{H}$ , kde  $i$ -tý sloupec matice  $\mathbf{W}$  značíme  $w_i$  a  $i$ -tý řádek matice  $\mathbf{H}$  značíme  $h_i$ . Masku pro  $i$ -tou složku značíme  $M_i$ .

$$\mathbf{WH} = \underbrace{w_1 \cdot h_1}_{M_1} + \underbrace{w_2 \cdot h_2}_{M_2} + \dots + \underbrace{w_r \cdot h_r}_{M_r}$$

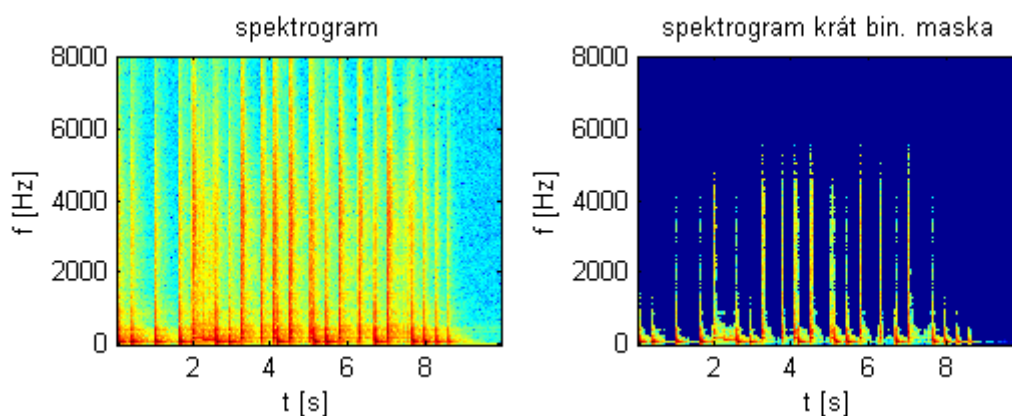
Maska nemusí být jen pro jednu složku. Lze vytvořit více masek pro jednotlivé složky a ty pak sečíst. Před samotnou aplikací masky na spektrogram ji lze ještě různými způsoby transformovat. Jedním z nich je převod na binární masku.

### 4.1.1 Binární maska

Binární maska obsahuje pouze hodnoty 0 nebo 1. Pro transformaci masky  $M$  na binární masku  $M_{BIN}$  stanovíme určitou mezní hodnotu  $lim$ . Pak binární maska

$$M_{BIN} = \begin{cases} 0 & \text{pro } M \leq lim \\ 1 & \text{pro } M > lim \end{cases} \quad (4.1)$$

Aplikováním binární masky na spektrogram se pak zachovají ty hodnoty spektrogramu, které jsou na stejných pozicích jako jedničky v binární masce. Ostatní hodnoty se vynulují. Úpravy spektrogramu provedené binární maskou ukazuje obrázek (4.3).

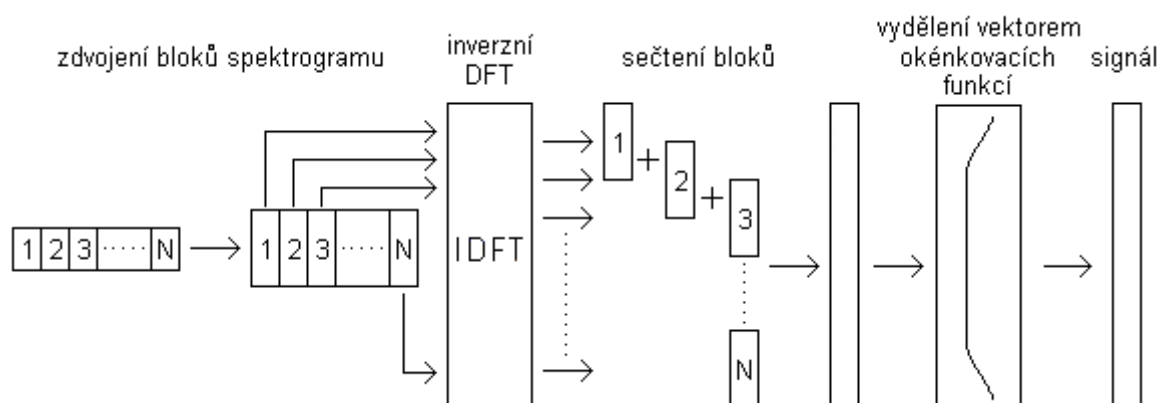


Obrázek 4.3: Aplikace binární masky na spektrogram

## 4.2 Inverzní spektrogram

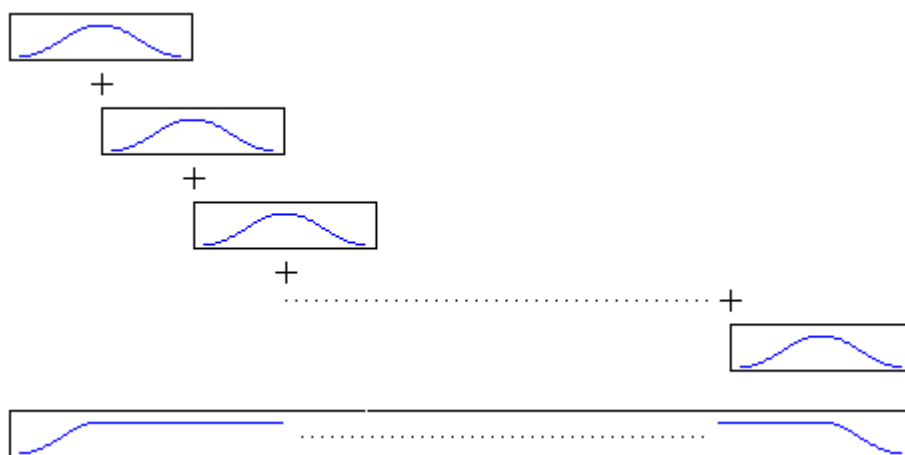
Separaci jednotlivých složek provedeme aplikací masek na spektrogram zpracovávaného signálu. Samotnou separaci tedy provádíme ve frekvenční oblasti. Spektrogram upravený maskou, který představuje jednu nebo více složek signálu, je pak třeba převést zpět do časové oblasti.

K převodu spektrogramu do časové oblasti je nutné znát parametry, jak byl spektrogram vypočítán. Převod spektrogramu do časové oblasti graficky znázorňuje obrázek (4.4).



Obrázek 4.4: Grafické znázornění převodu spektrogramu do časové oblasti

Hodnoty spektrogramu jsou polovina výsledků DFT (kapitola 3.2.4). Nejprve je tedy nutné každý blok spektrogramu zdvojit tak, aby měl strukturu jako původní výsledek DFT. Na takto upravené bloky je následně aplikována IDFT. Výsledky IDFT jsou pak poskládány za sebe, přičemž části bloků, které se překrývají, se sečtou. Velikost překrývání je stejná jako při výpočtu spektrogramu. Na konci se výsledný signál ještě vydělí pomocným vektorem, který je složen z okénkovacích funkcí. Tento pomocný vektor se získá podobně jako signál, ale jednotlivé bloky zde představují okénkovací funkci. Obrázek (4.5) znázorňuje určení pomocného vektoru okénkovacích funkcí [12].



Obrázek 4.5: Určení pomocného vektoru okénkovacích funkcí

Pokud se na spektrogramu provedou nějaké úpravy, může se stát, že v časové oblasti budou některé amplitudy mimo normalizovaný rozsah  $\langle -1; 1 \rangle$ . Pak je vhodné ještě upravit signál tak, aby byl v normalizovaném rozsahu.



## 4.3 Vytvoření stereo signálu

Vytvoření stereo signálu je v celém tomto procesu posledním krokem. Když už máme k dispozici jednotlivé oddělené složky, stačí je jen rozdělit do různých kanálů.

Rozklad pomocí NMF je ale pouze přibližný, proto není příliš vhodné vytvořit stereofonní signál jen z oddělených složek. Mohlo by tak dojít k celkovému poklesu kvality. Vhodnější je rozmístit jednotlivé složky do kanálů a k nim ještě přičíst zbytkový signál. Amplitudový spektrogram zbytkového signálu získáme odečtením aproximace  $\mathbf{WH}$  od matice  $\mathbf{V}$  představující spektrogram zpracovávaného signálu. Dostaneme tak v podstatě to, co se nepodařilo oddělit. Je vhodné tento zbytkový signál nechat jako monofonní, proto ho přičteme do všech kanálů.

Pořadí rozmístění jednotlivých složek do kanálů je obecně nejednoznačné. Záleží na zpracovávaném signálu a na inicializaci algoritmu. Rozmístění složek je pak třeba přizpůsobit dosaženým výsledkům rozkladu.

Není ovšem nutné promítnout do výsledného stereofonního signálu všechny oddělené složky. Pokud je ve zpracovávaném signálu obsažený šum, může se povést pomocí NMF tento šum oddělit a některé složky pak mohou obsahovat právě jen šum. Vynecháním těchto složek můžeme dosáhnout vyšší kvality výsledného signálu.

## Kapitola 5

# Měření kvality rozkladu – SIR a SDR

Pro určení kvality rozkladu do jednotlivých složek pomocí NMF používáme hodnoty *signal to interference ratio* (SIR) a *signal to distortion ratio* (SDR). Pro přesný výpočet hodnot SIR a SDR je ale nutné mít původní složky obsažené v signálu zvlášť.

V praxi ale není možné ke všem rozloženým audio signálům vypočítat přesné SIR a SDR, protože původní složky zvlášť k dispozici nejsou. Existují ale algoritmy, kterými lze tyto hodnoty odhadnout.

Přesný výpočet SIR a SDR lze provést jen v experimentální úloze, kde nejprve jednotlivé složky uměle smícháme a následně pomocí NMF rozložíme. V následující kapitole popisujeme postup přesného výpočtu SIR a SDR pro experimentální úlohu.

## 5.1 Postup výpočtu

Nejprve smícháme jednotlivé složky

$$X = S_1 + S_2 + \dots + S_r,$$

kde  $S_i$  jsou jednotlivé složky a  $X$  představuje součet všech složek. Pro signál  $X$  vypočítáme spektrogram a na něj aplikujeme NMF. Výsledky NMF použijeme k vytvoření binárních masek  $M_1 \dots M_r$ , přičemž

$$\begin{aligned} M_1 &\rightarrow \text{maska pro } S_1 \\ M_2 &\rightarrow \text{maska pro } S_2 \\ &\vdots \\ M_r &\rightarrow \text{maska pro } S_r \end{aligned}$$

Dále provedeme separaci postupným aplikováním masek na spektrogram signálu  $X$  a takto upravené spektrogramy pak převedeme zpět do časové oblasti. Vzniknou tak oddělené složky

$$\begin{aligned} V_1 &= \text{ispecgram}(M_1 \cdot X) \\ V_2 &= \text{ispecgram}(M_2 \cdot X) \\ &\vdots \\ V_r &= \text{ispecgram}(M_r \cdot X) \end{aligned}$$

kde  $\text{ispecgram}()$  je funkce, která převádí spektrogram zpět do časové oblasti. Nelze ale očekávat, že se separace podaří úplně dokonale. Pravděpodobně tedy bude platit, že

$$S_i \neq V_i.$$

Funkce  $\text{ispecgram}()$  je okénkovací IDFT a pro definování SIR a SDR je důležité, že se jedná o lineární operaci. Pro součet vyseparovaných složek  $Y$  tedy platí

$$\begin{aligned} Y &= \text{ispecgram}(M \cdot X) \\ &= \text{ispecgram}(M \cdot S_1) + \text{ispecgram}(M \cdot S_2) + \dots + \text{ispecgram}(M \cdot S_r) \end{aligned}$$

Obsah oddělené složky  $V_i$  lze rozepsat jako

$$\begin{aligned}
V_i &= \text{ispecgram}(M_i \cdot X) = \text{ispecgram}(M_i \cdot (S_1 + S_2 + \dots + S_r)) \\
&= \text{ispecgram}(M_i \cdot S_1 + M_i \cdot S_2 + \dots + M_i \cdot S_r) \\
&= \underbrace{\text{ispecgram}(M_i \cdot S_1)}_{\text{signal}} + \underbrace{\text{ispecgram}(M_i \cdot S_2) + \dots + \text{ispecgram}(M_i \cdot S_r)}_{\text{interference}}
\end{aligned}$$

Jak vyjadřuje vztah (5.1), SIR je poměr energie *signálu* a energie *interference*.

$$SIR = \frac{\text{mean}([signal]^2)}{\text{mean}([interference]^2)} \quad (5.1)$$

Podle vztahu (5.2) se SDR určí jako poměr energie *signálu* a energie *distortion*. *Distortion* je definováno jako  $S_i - M_i \cdot S_i$ .

$$SDR = \frac{\text{mean}([signal]^2)}{\text{mean}([distortion]^2)} \quad (5.2)$$

Hodnoty SIR a SDR se mohou počítat v decibelech. Výpočet SIR a SDR v decibelech uvádí vztahy (5.3) a (5.4).

$$SIR_{dB} = 10 \cdot \log_{10} \left( \frac{\text{mean}([signal]^2)}{\text{mean}([interference]^2)} \right) \quad (5.3)$$

$$SDR_{dB} = 10 \cdot \log_{10} \left( \frac{\text{mean}([signal]^2)}{\text{mean}([distortion]^2)} \right) \quad (5.4)$$

# Kapitola 6

## Praktická aplikace

V této kapitole popisujeme aplikaci NMF na konkrétní monofonní zvukové signály a porovnáváme výsledky dosažené s různým nastavením. Z výsledků rozkladu monofonního signálu pak vytváříme signál stereofonní.

Každý analyzovaný signál rozložíme pomocí NMF se třemi objektivními funkcemi (Euklidovská vzdálenost (2.5), KL divergence (2.6) a IS divergence (2.7)). Různě nastavujeme redukovanou dimenzi  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$  (počet složek, do kterých se provádí rozklad). Rozklady provádíme pro  $r = 3, 4, 5$  a  $6$ . Pro každý výpočet jsme provedli 1000 iterací NMF algoritmu.

Aplikujeme postup, který jsme popsali v kapitole 4. Pro samotné výpočty používáme některé funkce MATLABu a vlastní implementaci popsanou v příloze B.

V první části rozkládáme signál, který je uměle vytvořen jako součet tří složek. Jelikož máme k dispozici jednotlivé složky zvlášť, můžeme určovat kvalitu rozkladu pomocí hodnot SIR (5.3) a SDR (5.4). Ve druhé části zkoumáme výsledky rozkladu zašuměného zvukového signálu a testujeme, zda se podaří složky oddělit od šumu. Dále pak zpracováváme skutečnou nahrávku. Aplikujeme NMF rozklad na píseň *Hledaný muž* od kapely *Taxmeni*.

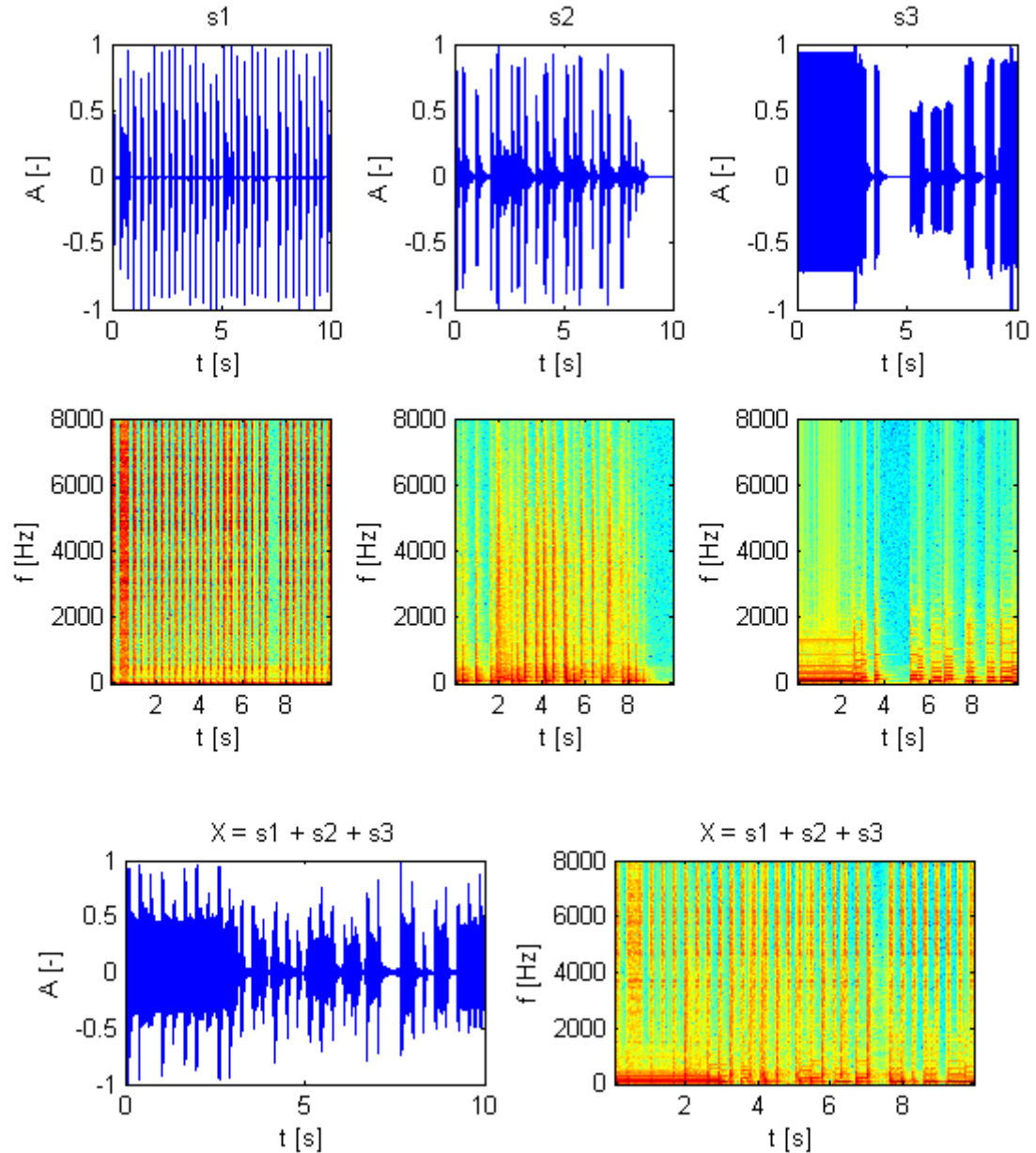
Aby se daly výsledky lépe srovnávat, inicializaci matic  $\mathbf{W}$  a  $\mathbf{H}$  pro stejnou redukovanou dimenzi  $r$  jsme provedli vždy stejnými hodnotami. Mezní hodnotu pro vytvoření binárních masek jsme zvolili 0,5.

Výpočty jsme prováděli na počítači s procesorem AMD Athlon 64 X2 3600+ a s operační pamětí 2 GB RAM.

## 6.1 Rozklad signálu složeného ze tří složek

Nejprve vytvoříme směs tří složek  $X = S_1 + S_2 + S_3$ . Složka  $S_1$  obsahuje hru na činely, složka  $S_2$  hru na bicí a ve složce  $S_3$  jsou basy. Vzorkovací frekvence signálu je 16000 Hz, délka signálu je 10 sekund.

Obrázek (6.1) zobrazuje jednotlivé složky  $S_1$ ,  $S_2$ ,  $S_3$  a jejich součet  $X$ .



Obrázek 6.1: Složky signálu  $S_1$ ,  $S_2$ ,  $S_3$  a součet  $X$  v časové oblasti a jejich spektrogramy

Spektrogramy byly vypočítány s délkou bloků 1024 vzorků, použita byla Hammingova okénkovací funkce a velikost překrývání bloků byla nastavena na 1000 vzorků. K výpočtu spektrogramu jsme použili funkci MATLABu *spectrogram.m*.

V následujících tabulkách jsme popsali obsahy vyseparovaných složek podle subjektivního hodnocení poslechem. Původní složku značíme  $S_i$  a vyseparovanou složku  $V_i$ . Objektivní funkci značíme  $EUC$  pro Euklidovskou vzdálenost,  $KL$  pro Kullback-Leibler divergenci a  $IS$  pro Itakura-Saito divergenci. Výsledky rozkladů a výsledný stereofonní signál jsou k dispozici ve formátu *wav* na přiloženém CD.

Tabulka 6.1: Výsledky rozkladu do tří složek

	$NMF - EUC$	$NMF - KL$	$NMF - IS$
$V_1$	část všech složek	bicí ( $S_2$ ) a činely ( $S_1$ )	činely ( $S_1$ )
$V_2$	část všech složek	část basů ( $S_3$ )	bicí ( $S_2$ )
$V_3$	část všech složek	bicí ( $S_2$ ) a část basů ( $S_3$ )	basy ( $S_3$ )

Tabulka 6.2: Výsledky rozkladu do čtyř složek

	$NMF - EUC$	$NMF - KL$	$NMF - IS$
$V_1$	činely ( $S_1$ ) a část bicích ( $S_2$ )	část basů ( $S_3$ )	činely ( $S_1$ )
$V_2$	činely ( $S_1$ ) a část bicích ( $S_2$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	bicí ( $S_2$ )
$V_3$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	basy ( $S_3$ )
$V_4$	část všech složek	činely ( $S_1$ ) a bicí ( $S_2$ )	činely ( $S_1$ )

Tabulka 6.3: Výsledky rozkladu do pěti složek

	$NMF - EUC$	$NMF - KL$	$NMF - IS$
$V_1$	část basů ( $S_3$ )	část basů ( $S_3$ )	činely ( $S_1$ )
$V_2$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	část basů ( $S_3$ )	část basů ( $S_3$ )
$V_3$	činely ( $S_1$ ) a bicí ( $S_2$ )	činely ( $S_1$ )	činely ( $S_1$ )
$V_4$	část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	bicí ( $S_2$ )
$V_5$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	bicí ( $S_2$ )	bicí ( $S_2$ ) a basy ( $S_3$ )

Tabulka 6.4: Výsledky rozkladu do šesti složek

	$NMF - EUC$	$NMF - KL$	$NMF - IS$
$V_1$	část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )
$V_2$	část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	část bicích ( $S_2$ )
$V_3$	činely ( $S_1$ ) a bicí ( $S_2$ )	činely ( $S_1$ )	činely ( $S_1$ )
$V_4$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	bicí ( $S_2$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )
$V_5$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ )	činely ( $S_1$ )
$V_6$	část činelů ( $S_1$ ) a část basů ( $S_3$ )	činely ( $S_1$ )	část bicích ( $S_2$ ) a část basů ( $S_3$ ))



### 6.1.1 Zhodnocení výsledků

NMF rozklad s použitím Euklidovské vzdálenosti dopadl při tomto testu nejhůře. Téměř vždy se stalo, že do výsledných složek se dostaly pouze části původních složek, a to navíc různě pomíchané. Výsledky se různě měnily s nastavením redukované dimenze  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$ . Pouze při rozkladu do šesti složek se podařilo oddělit basy ( $S_3$ ). Činely ( $S_1$ ) a bicí ( $S_2$ ) se nepovedlo zcela oddělit ani jednou. Několikrát se ve výsledných složkách objevila jejich směs.

Lépe dopadly výsledky s KL divergencí. Někdy ale byly velmi podobné jako výsledky s Euklidovskou vzdáleností. Opět se stávalo, že ve výsledných složkách byly jen části původních složek smíchané s ostatními. Výsledky byly také ovlivněny nastavením redukované dimenze  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$ . Při rozkladu do pěti složek se podařilo dobře oddělit všechny tři složky. Rozkladem do šesti složek se povedlo oddělit činely ( $S_1$ ) a bicí ( $S_2$ ). Stejně jako u rozkladu s Euklidovskou vzdáleností se stávalo, že některé výsledné složky obsahovaly směs činelů ( $S_1$ ) a bicích ( $S_2$ ).

Nejlepší výsledky v tomto testu poskytla NMF s IS divergencí. Již při rozkladu do tří složek byly výsledky velmi dobré a podařilo se oddělit všechny tři složky. S vyšší redukovanou dimenzí  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$  se výsledky také měnily a někdy se pouze části původních složek promítly do výsledných složek. Rozkladem do čtyř složek se ještě povedlo oddělit všechny tři složky, ale po rozkladu do pěti složek byly oddělené jen činely ( $S_1$ ) a bicí ( $S_2$ ). Basy ( $S_3$ ) se různě pomíchalý s ostatními složkami. Při rozkladu do šesti složek se povedlo oddělit už jen činely ( $S_1$ ).

Tabulka (6.5) obsahuje hodnoty SIR a SDR úspěšných výsledků. K výsledkům je uvedená objektivní funkce a hodnota redukované dimenze  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$ . Uvedené jsou jen ty výsledky, ve kterých se povedlo oddělit alespoň jednu celou složku. Pokud se některá složka nepodařila oddělit, je buňka tabulky proškrtnutá.

Jak je ale patrné z tabulky (6.6), NMF rozklad s Euklidovskou vzdáleností měl nejmenší výpočetní náročnost. Rozklad s KL divergencí trval oproti Euklidovské vzdálenosti průměrně  $2,6\times$  déle a s IS divergencí průměrně  $4,2\times$  déle.

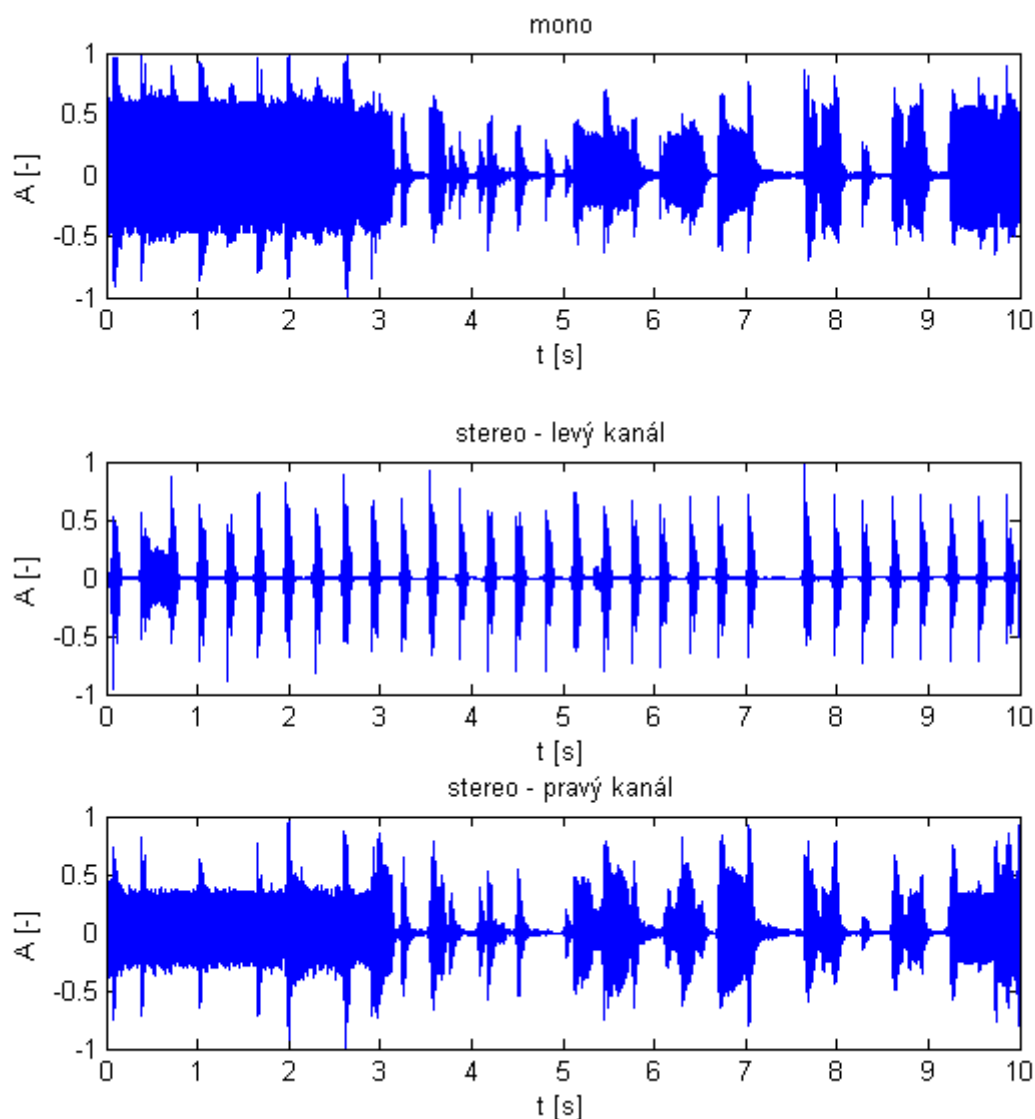
Tabulka 6.5: Hodnoty SIR a SDR rozkladu signálu složeného ze tří složek

	<i>SIR</i>			<i>SDR</i>		
<i>NMF</i>	$V_1$	$V_2$	$V_3$	$V_1$	$V_2$	$V_3$
$IS_{r=3}$	-1,0041	6,9438	7,8376	-1,6483	-0,7236	28,0652
$IS_{r=4}$	0,0979	8,1159	7,8291	-1,5970	-0,8839	27,7106
$KL_{r=5}$	-1,658	6,0699	6,0965	0,3632	-2,4642	13,1801
$IS_{r=5}$	-1,0565	5,7655	—	-2,0704	-4,525	—
$EUC_{r=6}$	—	—	8,0968	—	—	8,5331
$KL_{r=6}$	-2,8668	-2,2347	—	-0,1586	-3,5886	—
$IS_{r=6}$	0,0368	—	—	-2,3757	—	—

Tabulka 6.6: Časy výpočtů rozkladu signálu složeného ze tří složek

	<i>NMF – EUC</i>	<i>NMF – KL</i>	<i>NMF – IS</i>
$r=3$	1 min 28 s	4 min 46 s	8 min 13 s
$r=4$	2 min 6 s	5 min 46 s	9 min 16 s
$r=5$	2 min 31 s	6 min 21 s	9 min 45 s
$r=6$	2 min 39 s	6 min 25 s	9 min 52 s

Obrázek (6.2) zobrazuje původní monofonní a výsledný stereofonní signál.



Obrázek 6.2: Původní monofonní a výsledný stereofonní signál v časové oblasti

## 6.2 Rozklad zašuměného signálu

Zpracovávaný signál je podobný signálu z kapitoly 6.1. Je také složen ze tří složek, kde složka  $S_1$  obsahuje hru na činely, složka  $S_2$  hru na bicí, ale ve složce  $S_3$  je šum.

Spektrogramy byly vypočítány se stejným nastavením, tj. délka bloků 1024 vzorků, použita byla Hammingova okénková funkce a překrytí bloků bylo nastaveno na 1000 vzorků.

## 6.2.1 Zhodnocení výsledků

Oddělení složek ze zašuměného signálu dopadlo v tomto testu úspěšně. Zhodnocujeme pouze ty výsledky, ve kterých nebyl při poslechu žádný šum znatelný.

Bicí ( $S_2$ ) se podařilo oddělit téměř vždy. Pouze při rozkladu do tří složek s Euklidovskou vzdáleností byly zašuměné všechny výsledné složky. Ve všech dalších rozkladech byly ale bicí ( $S_2$ ) velmi dobře odděleny.

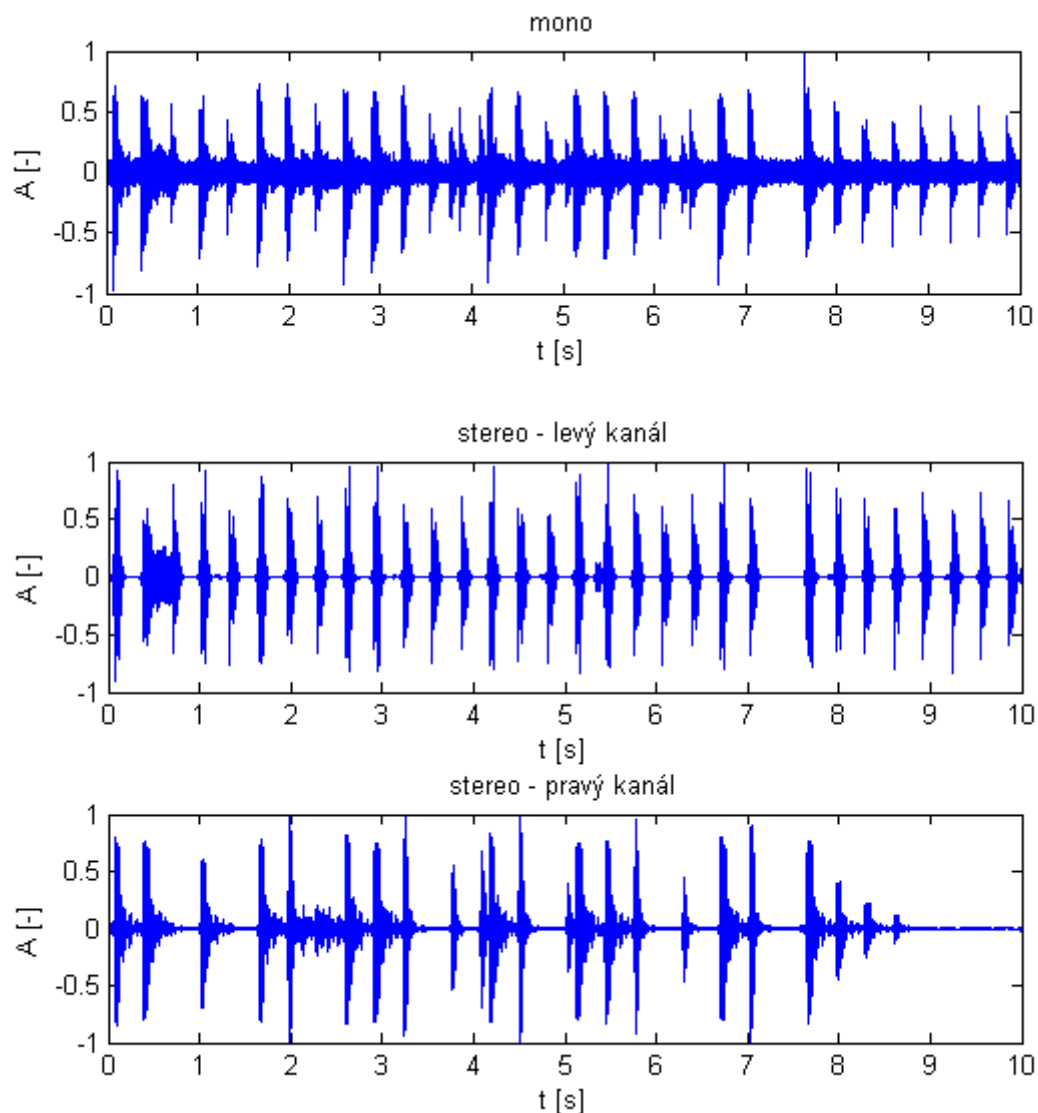
Činely ( $S_1$ ) se povedlo oddělit s IS divergencí při rozkladu do tří složek a při rozkladu do čtyř složek s KL a IS divergencí. Ostatní výsledky znatelně obsahovaly šum.

V tabulce (6.7) jsou uvedené hodnoty SIR a SDR jen pro činely ( $S_1$ ) a bicí ( $S_2$ ), protože cílem tohoto testu nebylo oddělit šum ( $S_3$ ) do samostatné složky. Časy výpočtů byly přibližně stejně dlouhé jako v kapitole 6.1. Výsledky rozkladů a výsledný stereofonní signál jsou k dispozici ve formátu *wav* na příloženém CD.

Tabulka 6.7: Hodnoty SIR a SDR rozkladu zašuměného signálu

	<i>SIR</i>		<i>SDR</i>	
<i>NMF</i>	$V_1$	$V_2$	$V_1$	$V_2$
$KL_{r=3}$	—	7,9655	—	-0,0831
$IS_{r=3}$	-1,206	10,2236	-0,1964	8,9893
$EUC_{r=4}$	—	8,8578	—	-1,4034
$KL_{r=4}$	-1,8135	7,7137	5,1370	0,32
$IS_{r=4}$	-2,4902	10,3826	-1,1261	8,4869
$EUC_{r=5}$	—	8,2673	—	-1,7040
$KL_{r=5}$	—	8,6633	—	-0,7255
$IS_{r=5}$	—	7,5542	—	-1,4404
$EUC_{r=6}$	—	8,6477	—	-1,6592
$KL_{r=6}$	—	8,3452	—	-2,3899
$IS_{r=6}$	—	10,6405	—	9,5873

Na obrázku (6.3) je zobrazen původní monofonní a výsledný stereofonní signál.



Obrázek 6.3: Původní monofonní a výsledný stereofonní signál v časové oblasti

### 6.3 Rozklad reálné nahrávky

Z písně *Hledaný muž* od kapely *Taxmeni* jsme zpracovávali prvních 10 sekund. Jelikož tato nahrávka byla stereofonní, pro testování NMF rozkladu jsme ji nejprve převedli na monofonní. Vzorkovací frekvence signálu byla 44100 Hz.

Spektrogram jsme vypočítali se stejně jako v předchozích testech (délka bloků 1024 vzorků, Hammingova okénkovácí funkce a překrytí bloků 1000 vzorků).

### 6.3.1 Zhodnocení výsledků

Výsledky zhodnocujeme opět subjektivně poslechem. Všechny NMF rozklady s použitými objektivními funkcemi měly podobný charakter. Složky, které byly dobře odděleny s nižší redukovanou dimenzí  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$ , se obvykle objevily ve výsledných složkách téměř stejně i s vyšší redukovanou dimenzí  $r$ . V dalších složkách se pak objevoval různě rozdělený zbytek signálu.

Rozkladem pomocí NMF s Euklidovskou vzdáleností se s každým nastavením redukované dimenze  $r$  objevil ve výsledné složce doprovod na bicí. Zbytek signálu se pak vždy rozdělil do složek, ve kterých převažovaly buď nižší nebo vyšší frekvence.

Zajímavější výsledky poskytla NMF s KL divergencí. Ve výsledných složkách se objevoval opět doprovod na bicí, ale navíc i kytara. Podobně jako u výsledků s Euklidovskou vzdáleností se pak zbytek signálu rozděloval do složek, ve kterých byly výraznější nižší nebo vyšší frekvence.

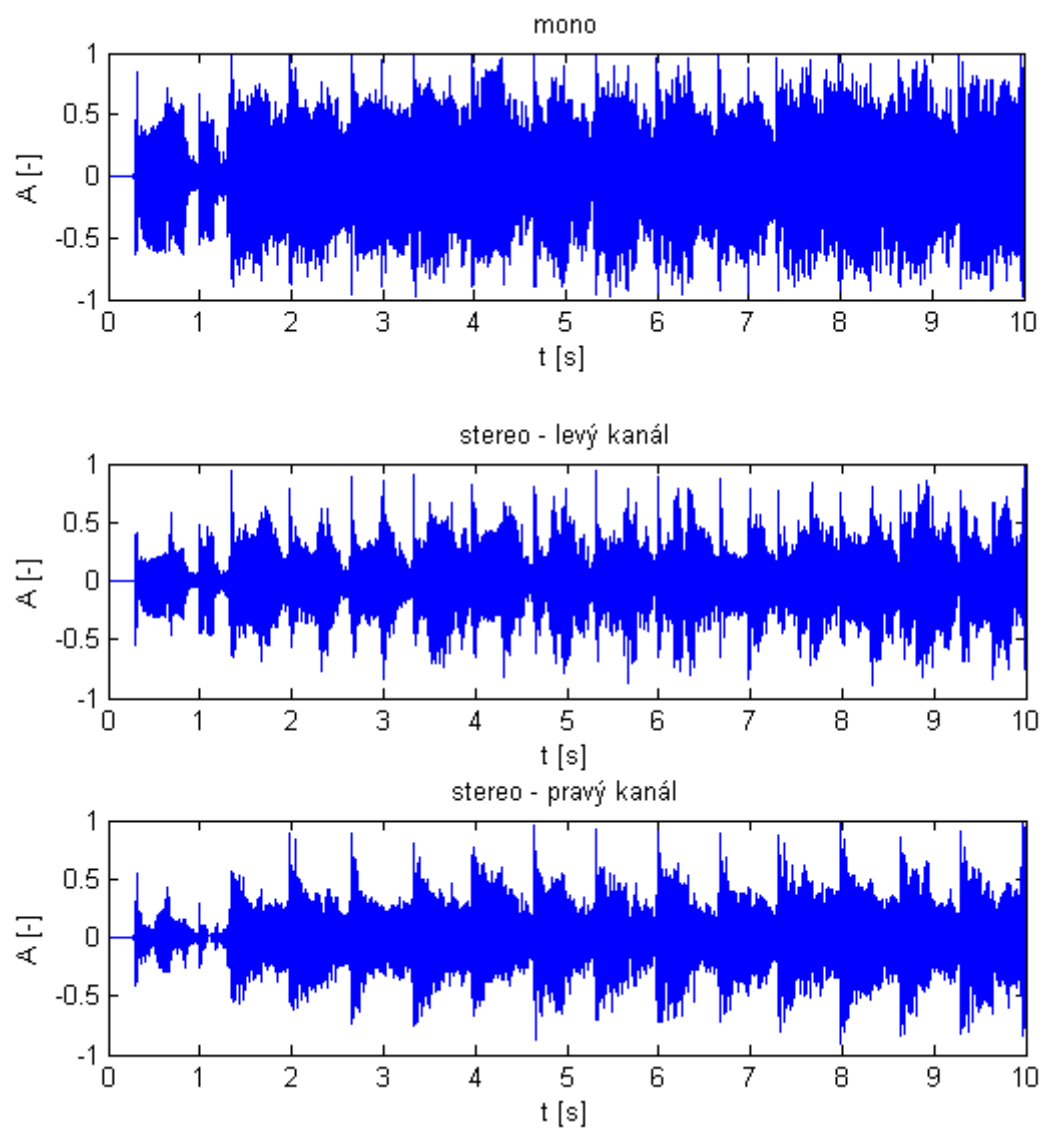
Jiných výsledků jsme dosáhli použitím NMF s IS divergencí. S každým nastavením redukované dimenze  $r$  byla ve výsledcích složka, ve které byla obsažena většina signálu. Dále pak složka s hloubkami a jako u KL divergence složka s kytarou. Při zvyšování redukované dimenze  $r$  se pak kytara rozdělovala mezi více složek. Složka s většinou signálu a složka s hloubkami zůstávaly nezměněné.

Tabulka (6.8) uvádí časy výpočtů rozkladů. NMF s Euklidovskou vzdáleností měla opět nejmenší výpočetní náročnost, NMF s KL divergencí trvala oproti Euklidovské vzdálenosti průměrně  $2,6\times$  déle a IS divergencí průměrně  $4,3\times$  déle. Výsledky rozkladů a výsledný stereofonní signál jsou k dispozici ve formátu *wav* na přiloženém CD.

Tabulka 6.8: Časy výpočtů rozkladu části písně *Hledaný muž* od kapely *Taxmeni*

	<i>NMF – EUC</i>	<i>NMF – KL</i>	<i>NMF – IS</i>
$r=3$	4 min 8 s	13 min 23 s	22 min 39 s
$r=4$	5 min 49 s	16 min 6 s	25 min 35 s
$r=5$	6 min 52 s	17 min 35 s	28 min 13 s
$r=6$	7 min 25 s	18 min 12 s	29 min 15 s

Obrázek (6.4) zobrazuje původní monofonní a výsledný stereofonní signál.



Obrázek 6.4: Původní monofonní a výsledný stereofonní signál v časové oblasti

# Kapitola 7

## Závěr

Během testování NMF algoritmu se ukázalo, že celkovou kvalitu rozkladu lze ovlivnit několika způsoby. Jedná se především o nastavení redukované dimenze  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$ , dále záleží na zvolené objektivní funkci, na parametrech, se kterými byl vypočítán spektrogram, a i na mezní hodnotě binární masky. Výsledky byly ovlivněné i inicializací matic  $\mathbf{W}$  a  $\mathbf{H}$ .

Vhodné nastavení redukované dimenze  $r$  matic  $\mathbf{W}$  a  $\mathbf{H}$  je pro NMF rozklad velmi důležité a výrazně ovlivní kvalitu celého procesu rozkladu. Nelze však předem jednoznačně určit, s jakou hodnotou redukované dimenze  $r$  se dosáhne nejlepšího výsledku. Během našich testů se ukázalo, že v určitých případech je vhodnější nastavit redukovanou dimenzi  $r$  spíše na nižší hodnotu a provést tak rozklad do méně složek. Rozkladem do více složek se pak někdy stávalo, že výsledné složky byly různě pomíchané a objevovaly se v nich pouze části původních složek smíchané s ostatními. Mnohdy vyšší redukovaná dimenze  $r$  nepřinesla žádné lepší výsledky. Zvyšování redukované dimenze  $r$  také způsobí vyšší výpočetní náročnost. Samozřejmě ale pro některé signály může být vhodnější provést rozklad do více složek. Hledání nejvhodnější hodnoty redukované dimenze  $r$  jsme prováděli experimentálně.

Výsledky byly výrazně ovlivněny i zvolenou objektivní funkcí. Rozklad signálu složeného ze tří složek z kapitoly 6.1 a zašuměného signálu z kapitoly 6.2 dopadl nejlépe pomocí NMF s IS divergencí. Rozklad písně *Hledaný muž* od kapely *Taxmeni* z kapitoly 6.3 dopadl nejlépe pomocí NMF s KL divergencí. Zajímavé výsledky zde poskytla i NMF s IS divergencí. Rozklady s Euklidovskou vzdáleností měly ale vždy nejmenší výpočetní náročnost.



Kvalita celého NMF rozkladu velmi závisí na parametrech, podle kterých byl vypočítán spektrogram zpracovávaného signálu. Záleží především na délce bloků, na okénkovací funkci a na velikosti překrývání bloků. V našich testech jsme používali délku bloků 1024 vzorků a překrytí bloků 1000 vzorků. Spektrogram 10 sekundového signálu s vzorkovací frekvencí 44100 Hz byl pak matice o rozměrech  $513 \times 18333$ , což při NMF rozkladu představuje relativně vysokou výpočetní náročnost. NMF rozklad jsme testovali i pro spektrogramy s menším překrýváním bloků (například 128, 256, 512, 640, 768 a 896 vzorků, při délce bloků 1024 vzorků). Ve výsledných složkách pak ale bylo i výrazně obsaženo praskání. Celková kvalita tak byla velmi nízká, proto jsme používali spektrogramy s větším překrýváním bloků. S velkým překrýváním bloků pak nebylo ve výsledcích žádné praskání znatelné. Výsledky byly srovnatelné i s délkou bloků 512 vzorků, opět ale muselo být nastaveno velké překrývání bloků. Záleželo i na volbě okénkovací funkce. Jelikož jsme pak spektrogramy převáděli zpět do časové oblasti, bylo vhodné použít okénkovací funkci, která nikde nenabývá nulové hodnoty. Násobením nulou by totiž docházelo ke ztrátám informací. S různými okénkovacími funkcemi jsme dosahovali i různých výsledků. Výraznější rozdíly byly ve výsledcích, jen když se i výrazně lišily okénkovací funkce. Rozklady spektrogramů, ve kterých byly použity podobné okénkovací funkce, měly i podobné výsledky. Výrazné rozdíly byly například při porovnání rozkladů spektrogramů, ve kterých byla použita obdélníková okénkovací funkce nebo Hammingova okénkovací funkce. V našich rozkladech jsme nejčastěji používali Hammingovu okénkovací funkci.

Dalším ovlivňujícím faktorem výsledku je mezní hodnota při transformaci masky na binární. S vyšší mezní hodnotou bude maska obsahovat více nul a tak se do výsledku dostane menší část složky. Vhodným nastavením mezní hodnoty lze také dosáhnout lepších výsledků. Podobně jako hodnotu redukované dimenze  $r$  jsme i optimální mezní hodnotu binární masky hledali experimentálně.

Jelikož inicializaci výsledných matic  $\mathbf{W}$  a  $\mathbf{H}$  jsme prováděli náhodnými nezápornými čísly, pro různou inicializaci vycházely i různé výsledky. S různou inicializací si výsledky byly ale velmi podobné. Stávalo se jen, že výsledné složky byly v jiném pořadí, což je v souladu s vlastnostmi NMF rozkladu.

Záleží ale samozřejmě i na samotném zpracovávaném signálu. Může se stát, že se zpracovávaný signál nepodaří rozdělit do jednotlivých složek v požadované kvalitě. Všechny výše uvedené skutečnosti mají ale vliv na konečnou kvalitu rozkladu a je tak celá řada parametrů a nastavení, kterými lze výsledky zlepšit. Většinu parametrů a nastavení ale nelze určit optimálně předem, a tak experimentální hledání nejlepších výsledků může být velmi časově náročné.

Pro vytvoření stereofonního signálu z oddělených složek jsme již jen rozdělili jednotlivé složky do kanálů. K nim jsme ale ještě přičítali zbytkový signál (kapitola 4.3). Pořadí složek jsme vždy museli přizpůsobit výsledkům rozkladu. Aby dobře oddělené složky vynikly, obvykle jsme je nesmíchávali s ostatními složkami a raději jsme je nechávali v kanálu jen se zbytkovým signálem. Ostatní složky, nebo ty, které si byly při poslechu podobné, bylo výhodnější sečíst. Rozdělit do kanálů podobné složky by mohlo pak způsobit, že by se výsledný signál jevil více jako monofonní než stereofonní.

Metoda NMF poskytuje zajímavé možnosti nejen v oblasti zpracování zvukových signálů, ale také ve zpracování textu, obrazu, v zobrazovacích a diagnostických metodách v lékařství a v dalších, jak jsme uvedli v kapitole 2.4. Lze předpokládat, že pro NMF bude nalezeno uplatnění i v mnoha dalších odvětvích vědy a její výsledky by v budoucnu mohly být využívány i v běžném životě.

# Příloha A

## Obsah příloženého CD

`\Puvodni signaly\` – původní zpracovávané signály

`\Skripty\` – implementace skriptů popsaných v příloze B

`\Stereo\` – vytvořené stereofonní signály

`\Vysledky rozkladu\`

`\nahravka\` – Výsledky rozkladů reálné nahrávky z kapitoly 6.3

`\s12noise\` – Výsledky rozkladů zašuměného signálu z kapitoly 6.2

`\s123\` – Výsledky rozkladů signálu složeného ze tří složek z kapitoly 6.1

`bakalarska_prace.pdf` – tato práce v elektronické podobě

# Příloha B

## Manuál k implementaci

Implementace všech kódů jsme provedli a otestovali v MATLABu R2008a.

### B.1 NMF

Funkce *nmf.m* vykonává NMF algoritmus podle zadaných parametrů.

volání:  $[W \ H] = \text{nmf}(V, \text{eps}, \text{sekv}, \text{iterMax}, r, \text{dvr}, \text{varargin})$

Vstupy:

$V$  – vstupní matice nezáporných dat (absolutní hodnota spektrogramu)

$\text{eps}$ ,  $\text{sekv}$  – hodnoty pro test konvergence

$\text{iterMax}$  – maximální počet kroků

$r$  – redukovaná dimenze matic  $\mathbf{W}$  a  $\mathbf{H}$

$\text{dvr}$  – objektivní funkce

$\text{varargin}$  – inicializované matice  $\mathbf{W}$  a  $\mathbf{H}$  (nepovinné parametry)

Výstupy:

$W$ ,  $H$  – výsledné matice

Pokud se v parametru *varargin* nepředají inicializované matice  $\mathbf{W}$  a  $\mathbf{H}$ , provede se inicializace matic  $\mathbf{W}$  a  $\mathbf{H}$  náhodnými nezápornými čísly. Funkce končí a vrátí výsledky, pokud je splněna alespoň jedna ze dvou podmínek. Jednou je dosažení maximálního počtu kroků (parametr *iterMax*) a druhou podmínka konvergence (parametry *eps* a *sekv*).

Parametr *dvr* udává divergenci a její možné hodnoty jsou:

*euc* – Euklidovská vzdálenost

*kl* – Kullback-Leibler divergence

*is* – Itakura-Saito divergence

Test konvergence se provádí porovnáváním aktuálního a předchozího stavu matic **W** a **H**. Pro matici **W** se v každém kroku algoritmu vypočítá

$$W_{sum} = \sum_{i=1}^n \sum_{j=1}^r [ | W_{iter} - W_{iter-1} | ]_{i,j}$$

a pro matici **H**

$$H_{sum} = \sum_{i=1}^r \sum_{j=1}^m [ | H_{iter} - H_{iter-1} | ]_{i,j}$$

Pokud je hodnota  $W_{sum}$  menší než parametr *eps sekv*-krát po sobě, matice **W** se označí za zkonvergovanou. Podobným způsobem se testuje i matice **H**, a když zkonvergují obě matice, algoritmus končí a vrací výsledky.

## B.2 Maska

Funkce *binMask.m* vytvoří z výsledků NMF (matice **W** a **H**) binární masku pro složky *com*.

volání:  $[mask] = binMask(W, H, com, lim)$

Vstupy:

*W, H* – výsledky rozkladu NMF (matice **W** a **H**)

*com* – složky

*lim* – mezní hodnota

Výstupy:

*mask* – binární maska pro složky *com*

Maska pro  $i$ -tou složku se vytvoří maticovým vynásobením  $i$ -tého sloupce matice  $\mathbf{W}$  a  $i$ -tého řádku matice  $\mathbf{H}$ . Parametr *com* udává, pro které složky bude maska vytvořena (parametr *com* může být skalár nebo vektor). Pokud se má maska vytvořit pro více složek, výsledky maticového násobení sloupců matice  $\mathbf{W}$  a řádků matice  $\mathbf{H}$  se sečtou. Takto vytvořená maska se přetransformuje na binární. Všechny hodnoty větší než mezní hodnota (parametr *lim*) se nastaví na 1 a ostatní na 0.

## B.3 Inverzní spektrogram

Funkce *ispectrogram.m* převádí spektrogram do časové oblasti.

volání:  $[y] = \text{spectrogram}(S, \text{window}, \text{overlap})$

Vstupy:

$S$  – spektrogram

*window* – okénkovací funkce použitá při výpočtu původního spektrogramu

*overlap* – velikost překrytí bloků použitá při výpočtu původního spektrogramu

Výstupy:

$y$  – signál v časové oblasti

Pro převod vstupního spektrogramu (parametr  $S$ ) do časové oblasti je nutné znát parametry, se kterými byl původní spektrogram vypočítán. Jde o okénkovací funkci (parametr *window*, čímž je dána i velikost bloků) a velikosti překrytí jednotlivých bloků (parametr *overlap*). Nejprve se zdvojí bloky spektrogramu, aby měly strukturu původního výsledku DFT. Na ně se pak aplikuje IDFT pomocí funkce *ifft.m*, která je standardní součástí MATLABu. Bloky se pak skládají za sebe a části, které se překrývají, se sečtou. Na konci je ještě výsledek vydělen pomocným vektorem okénkovacích funkcí. Určení pomocného vektoru okénkovacích funkcí je popsáno v kapitole 4.2. Před dokončením je ještě signál normalizován (upraven do rozsahu  $\langle -1; 1 \rangle$ ).

## B.4 Uložení výsledků

Pro uložení výsledků ve formátu *wav* jsme implementovali funkce *writeResult.m* a *writeResults.m*. Funkce *writeResult.m* uloží do zadaného adresáře jednu vybranou složku a *writeResults.m* uloží do zadaného adresáře všechny složky.

### Uložení jedné složky

Funkce *writeResult.m* uloží do zadaného adresáře jednu oddělenou složku.

volání: *writeResult(S, window, overlap, W, H, s, lim, fs, path)*

Vstupy:

*S* – spektrogram zpracovávaného signálu

*window* – okénkovací funkce použitá při výpočtu původního spektrogramu

*overlap* – velikost překrytí bloků použitá při výpočtu původního spektrogramu

*s* – požadovaná složka

*W, H* – výsledky rozkladu NMF (matice **W** a **H**)

*lim* – mezní hodnota pro vytvoření binární masky

*fs* – vzorkovací frekvence zpracovávaného signálu

*path* – cesta k adresáři, do kterého se uloží výsledek

Funkce nemá žádnou návratovou hodnotu, výsledkem je uložení složky *s* do zadaného adresáře ve formátu *wav*. Nejprve se pro složku *s* vytvoří binární maska pomocí funkce *binMask.m* se zadanou mezní hodnotou (parametr *lim*). Pak se maska aplikuje na spektrogram (parametr *S*) a takto upravený spektrogram se převede do časové oblasti funkcí *ispectrogram.m*. Funkci *ispectrogram.m* je nutné předat parametry, podle kterých byl vypočítán původní spektrogram (parametry *window* a *overlap*). Výsledek v časové oblasti se uloží do zadaného adresáře (parametr *path*) funkcí *wavwrite.m*, která je standardní součástí MATLABu. Vytvořený soubor se jmenuje *sl.wav*.

## Uložení všech složek

Funkce *writeResults.m* uloží do zadaného adresáře všechny oddělené složky.

volání: *writeResults(S, window, overlap, W, H, lim, fs, path)*

Funkce *writeResults* pracuje na stejném principu jako *writeResult*. Ukladá ale do zadaného adresáře všechny složky. Pro každou složku se vytvoří binární maska se zadanou mezní hodnotou (parametr *lim*). Pak se maska aplikuje na spektrogram (parametr *S*) a upravený spektrogram se převede do časové oblasti funkcí *ispectrogram.m*. Výsledek se uloží do zadaného adresáře (parametr *path*) funkcí *wavwrite.m*. Tento postup se opakuje pro všechny složky. Názvy souborů jsou *x.wav*, kde *x* je číslo složky.

## B.5 Smíchání zvukových signálů

Funkce *makemix.m* vytvoří směs předaných zvukových signálů.

volání:  $[y] = \text{makemix}(sec, fs, varargin)$

Vstupy:

*sec* – požadovaná délka výsledného signálu [s]

*fs* – vzorkovací frekvence signálů

*varargin* – zvukové signály v časové oblasti

Výstupy:

*y* – výsledný signál

Signálů, které se mají smíchat, může být více. Proto jsou předány přes proměnný počet parametrů *varargin*. Pro správné fungování musí mít všechny předané signály stejnou vzorkovací frekvenci.

Pokud jsou předané signály stereofonní, sečtou se jejich kanály. Aby hlasitost všech signálů byla přibližně stejná, před samotným smícháním se ještě amplitudy všech signálů upraví do rozsahu  $\langle -1; 1 \rangle$ . Požadovaná délka výsledného signálu (parametr *sec*) může být větší, než je některý z předaných signálů, proto se porovnají



délky všech signálů a pokud je požadovaná délka příliš velká, vypíše se do konzole varování a výsledný signál bude mít pak největší možnou délku. Po smíchání signálů se ještě výsledek znormalizuje a ten se pak vrátí.

## B.6 Výpočet SIR

Funkce *sir.m* vypočítá hodnotu *signal to interference ratio* v decibelech pro zadanou složku.

volání:  $[sir] = sir(sp, mask, window, overlap, varargin)$

Vstupy:

*sp* – spektrogram složky *s* (původní složka před smícháním)

*mask* – maska pro složku *s*

*window* – okénkovací funkce použitá při výpočtu původního spektrogramu

*overlap* – velikost překrytí bloků použitá při výpočtu původního spektrogramu

*varargin* – spektrogramy všech dalších složek signálu

Výstupy:

*sir* – hodnota SIR v decibelech

Hodnota SIR v decibelech pro složku *sp* se vypočítá podle vztahu (5.3). Pro určení *interference* je třeba mít k dispozici všechny další složky obsažené v signálu. Jelikož jich může být více, jsou proto předány přes proměnný počet parametrů *varargin*. Příklad volání funkce pro výpočet SIR složky *s<sub>1</sub>*, kde signál je složen ze tří složek:

$[sir] = sir(sp1, mask1, sp2, sp3)$

## B.7 Výpočet SDR

Funkce *sdr.m* vypočítá hodnotu *signal to distortion ratio* v decibelech pro zadanou složku.

volání:  $[sdr] = sdr(sp, mask, window, overlap)$

Vstupy:

*sp* – spektrogram složky *s* (původní složka před smícháním)

*mask* – maska pro složku *s*

*window* – okénkovací funkce použitá při výpočtu původního spektrogramu

*overlap* – velikost překrytí bloků použitá při výpočtu původního spektrogramu

Výstupy:

*sdr* – hodnota SDR v decibelech

Hodnota SDR v decibelech pro zadanou složku se vypočítá podle vztahu (5.4).

# Literatura

- [1] D. D. Lee and S. H. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, vol. 401, no. 6755, pp. 788–791, 1999.
- [2] C. Févotte, N. Bertin, J.-L. Durrieu, *Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis*, Neural Comput., vol. 21, no. 3, pp. 793–830, 2009.
- [3] P. Paatero and U. Tapper, *Positive matrix factorization: A nonnegative factor model with optimal utilization of error-estimates of data values*, Environmetrics, vol. 5, no. 2, pp. 111–126, 1994.
- [4] D. D. Lee and S. H. Seung, *Unsupervised learning by convex and conic coding*, in Neural Information Processing Systems, Advances in (NIPS), pp. 515–521, 1996.
- [5] D. D. Lee and S. H. Seung, *Algorithms for non-negative matrix factorization*, in Neural Information Processing Systems, Advances in (NIPS), pp. 556–562, 2000.
- [6] N. Schmidt, *Single-channel source separation using non-negative matrix factorization*, Technical University of Denmark, 2008.
- [7] A. Cichocki, S.-I. Amari, R. Zdunek, R. Kompass, G. Hori, and Z. He, *Extended SMART algorithms for non-negative matrix factorization*, in Artificial Intelligence and Soft Computing, International Conference on (ICAISC), vol. 4029, pp. 548–562, 2006.

- [8] I. S. Dhillon and S. Sra, *Generalized nonnegative matrix approximations with Bregman divergences*, University of Texas at Austin, Department of Computer Sciences, Tech. Rep., 2005.
- [9] R. Kompass, *A generalized divergence measure for nonnegative matrix factorization*, Neural Computation, vol. 19, no. 3, pp. 780–791, 2007.
- [10] J. Nouza, *Signály a informace, studijní materialy* [online]. [cit. 6. 3. 2011].  
URL: <<https://www.ite.tul.cz/vyuka/mod/resource/view.php?id=9>>
- [11] *Spectrogram* [online]. [cit. 15. 4. 2011].  
URL: <<http://cnx.org/content/m0505/latest/>>
- [12] *Software manual for spectrogram project* [online]. [cit. 12. 4. 2011].  
URL: <[http://dsp.ucsd.edu/students/present-students/mik/specanalyzer/01\\_intro.htm](http://dsp.ucsd.edu/students/present-students/mik/specanalyzer/01_intro.htm)>